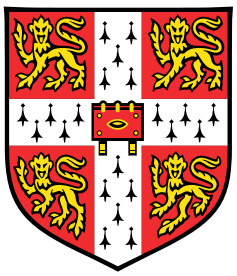
Domain adaptation for Neural Machine Translation



Danielle Saunders

Supervisor: Professor Bill Byrne

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Clare College November 2020

Dedicated to Bel.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, footnotes, tables and equations, and excluding the bibliography, and has fewer than 150 figures.

Danielle Saunders
November 2020

Acknowledgements

Before anything else, thanks must go to my supervisor, Bill Byrne, for guidance throughout the course of this PhD. Over the last four years he has highlighted new avenues for research while allowing me freedom to pursue my own ideas, provided constant insight and support, and given excellent and helpful feedback while preventing me from becoming too self-critical. I consider myself incredibly fortunate to have been his student.

I would also like to thank my collaborators. Felix Stahlberg provided helpful guidance and many enjoyable collaborations while we overlapped at Cambridge University, as well as being the primary developer of SGNMT, the inference framework I used or extended in many experiments. Marcus Tomalin, Stefanie Ullmann and Shauna Concannon first drew my attention to work on bias in translation, which became a major focus of my final year, and provided eye-opening perspectives via the Giving Voice to Digital Democracies project. Adrià de Gispert, Eva Hasler and Gonzalo Iglesias often pointed the way when I became lost in an experimental mire, and made me welcome for productive research placements at SDL, along with the rest of the SDL research group.

I was financially supported by Engineering and Physical Sciences Research Council grants EP/M508007/1 and EP/N509620/1. Many of my experiments were made possible with resources from the Cambridge Tier-2 system operated by the University of Cambridge Research Computing Service funded by EPSRC Tier-2 capital grant EP/P020259/1.

On a more personal note, I am incredibly lucky to have many excellent friends. Some have been a source of shared commiseration and joy through their own PhD journeys. All have been a source of happiness and support. The MIRTers in particular have brought companionship and laughter to a very isolated 2020.

My PhD would not have been the same without the love and support of my fiancée Isobel. She has stood by me with encouragement and patience throughout, even after bearing witness to my poster designs. I look forward to many more years together with far fewer deadlines.

My family has always supported me. Much of what led me to complete this thesis is owed to my mother, my father, and my brother Robin: my enjoyment of learning about new languages, my goal to become an engineer, and my enthusiasm for learning about anything and everything. Most of all they have helped me recognize the importance of perseverance.

Abstract

The development of deep learning techniques has allowed Neural Machine Translation (NMT) models to become extremely powerful, given sufficient training data and training time. However, such translation models struggle when translating text of a specific domain. A domain may consist of text on a well-defined topic, or text of unknown provenance with an identifiable vocabulary distribution, or language with some other stylometric feature. While NMT models can achieve good translation performance on domain-specific data via simple tuning on a representative training corpus, such data-centric approaches have negative side-effects. These include over-fitting, brittleness, and 'catastrophic forgetting' of previous training examples.

In this thesis we instead explore more robust approaches to domain adaptation for NMT. We consider the case where a system is adapted to a specified domain of interest, but may also need to accommodate new language, or domain-mismatched sentences. We explore techniques relating to data selection and curriculum, model parameter adaptation procedure, and inference procedure. We show that iterative fine-tuning can achieve strong performance over multiple related domains, and that Elastic Weight Consolidation can be used to mitigate catastrophic forgetting in NMT domain adaptation across multiple sequential domains. We develop a robust variant of Minimum Risk Training which allows more beneficial use of small, highly domain-specific tuning sets than simple cross-entropy fine-tuning, and can mitigate exposure bias resulting from domain over-fitting. We extend Bayesian Interpolation inference schemes to Neural Machine Translation, allowing adaptive weighting of NMT ensembles to translate text from an unknown domain.

Finally we demonstrate the benefit of multi-domain adaptation approaches for other lines of NMT research. We show that NMT systems using multiple forms of data representation can benefit from multi-domain inference approaches. We also demonstrate a series of domain adaptation approaches to mitigating the effects of gender bias in machine translation.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

BI	Bayesian Interpolation
BiRNN	Bidirectional Recurrent Neural Network
BLEU	Bilingual Evaluation Understudy
BOS	Beginning of Sentence
BP	Brevity Penalty
BPE	Byte Pair Encoding
CCG	Combinatory Categorial Grammar
CNN	Convolutional Neural Network
EOS	End of Sentence
EWC	Elastic Weight Consolidation
GEC	Grammatical Error Correction
GRU	Gated Recurrent Unit
ICS	Internal Covariate Shift
IDC	Ideographic Description Character
IS	Informative Source
LM	Language Model
LSTM	Long Short Term Memory

MERT Minimum Error Rate Training

MLE Maximum Likelihood Estimation

MRT Minimum Risk Training

MT Machine Translation

NLP Natural Language Processing

NMT Neural Machine Translation

OOV Out-of-Vocabulary

POS Part-of-speech

RL Reinforcement Learning

RNN Recurrent Neural Network

sBLEU Sentence-level BLEU

SMT Statistical Machine Translation

TER Translation Edit Rate

TF-IDF Term Frequency Inverse Document Frequency

VAE Variational Autoencoder

WMT Workshop on Statistical Machine Translation (2006-2015), Conference on Machine Translation (2016-)

Chapter 1

Introduction

1.1 Motivation

When automatically translating a sentence, a model that has previously encountered similar sentences is likely to produce a better translation. Likewise, a domain of sentences will be best translated by a Neural Machine Translation (NMT) system that is adapted to that domain. The focus of this thesis is on approaches to domain adaptation for NMT systems.

In this thesis, we interpret domain adaptation as any scheme intended to improve translations for a certain topic or genre of language. Examples include adapting model parameters with translations of sentences in the domain of interest, or constraining the model output to produce only vocabulary in the domain of interest.

In the definition of a domain we primarily follow Koehn and Knowles (2017), who state that a domain 'may differ from other domains in topic, genre, style, level of formality, etc.' However, we add two important caveats. Firstly we do not necessarily define a domain as 'a corpus from a specific source': while we find provenance useful for describing behaviour and reporting results, we do not consider it an exclusive domain marker. Secondly, in many practical cases the domain of test data is not known. We therefore draw a distinction in this thesis between work which is interested only in improved performance on a test set from a known domain, and work which aims to incorporate information from a certain training domain without loss of generalizability.

Most existing work on domain-specific language processing treats the domain of test language as known. This may be true in limited scenarios, such as shared tasks from the WMT machine translation conference where the topic or genre of text is pre-specified, or bespoke translation systems adapted to customer data. Many effective techniques have been developed for this known-domain scenario. For example, a recent and popular genre of work focuses on pre-training extremely large language models and then fine-tuning a set of

dedicated parameters on a smaller set of data for a fixed task (Devlin et al., 2019; Radford et al., 2019).

However, even in these known-domain settings, knowledge of domain label or provenance may not be as useful as it seems. A domain label applied to a corpus may not be representative of all sentences in that corpus. For example, this thesis might be labelled as a document from the technical or scientific domain, but that label would be completely unsuitable for sentences from the acknowledgements section. In a very general case, source sentences supplied to a freely available online translation system could come from any source and contain features of any domain.

Our focus is this more general case of multi-domain translation. Here a system may be required to translate language in a specified domain of interest, but it is not known that it will *only* be translating text relating to that domain. The translation system must therefore incorporate robustness to different domains.

This thesis therefore explores techniques for adaptation that can incorporate the benefits of a new domain without succumbing to brittleness, over-fitting, or general failure to successfully translate anything other than a chosen set of adaptation sentences. Techniques for multi-domain adaptation can involve data selection and curriculum, model parameter adaptation procedure, or choices made during inference. Moreover, we explore the extent to which some data-representation or data-selection focused schemes for improving machine translation can also be framed as individual 'domains' or benefit from multi-domain translation techniques.

1.1.1 Research questions

In this thesis we address five broad research questions around the topic of multi-domain adaptation for NMT. Here we introduce and motivate these questions.

How effective are data-centric approaches to NMT domain adaptation?

Early approaches to NMT adaptation involve training a general domain system, then fine-tuning the model on in-domain data (Luong and Manning, 2015). The adaptation data may be selected or generated in many different ways. It may be presented to the model as a distinct training phase, or it may be gradually incorporated into training, mixed with data from the general domain. We view these as purely data-centric approaches, in that the design choices revolve around the new data or its introduction.

Data-centric approaches remain the dominant approach to NMT adaptation, possibly due to their simplicity. We wish to assess their effectiveness for robust NMT domain adaptation,

especially in multi-domain scenarios. We also wish to identify any undesirable side-effects of such approaches.

Given an adaptation dataset, what training schemes might improve machine translation quality?

Approaches to NMT domain adaptation typically require some in-domain data. However, low-resource domains and language pairs may have very little data available for adaptation. In such cases, we wish to explore model parameter adaptation procedures that make the best use of the available data, while avoiding any undesirable side-effects of adapting to very small data-sets.

Can domain adaptation help when the test domain is unknown?

As described above, in a realistic scenario the domain of a new test sentence is unlikely to be predictable. The test set may also lack a convenient corresponding validation set. We wish to explore techniques that allow high-quality domain-specific machine translation in scenarios where the exact domain of a test sentence is not pre-determined.

Can changing data representation have similar effects to changing data domain?

A separate line of NMT research to domain adaptation concerns the way text is represented for translation purposes. Different levels of word or sub-word decomposition, or the presence of non-text tokens such as part-of-speech (POS) tags, may provide different translation benefits in various scenarios (Ding et al., 2019; Sennrich and Haddow, 2016). Drawing an analogy with multi-domain translation, we wish to investigate situations which benefit from more than one data representation, and techniques which benefit the resulting multi-representation NMT.

Can gender bias in NMT systems be mitigated by treating it as a domain?

Recent research has identified that NMT models exhibit unhelpful correlations between human-referent gendered terms. This behaviour is generally referred to as NMT gender bias (Alvarez-Melis and Jaakkola, 2017). We wish to know the extent to which this can be viewed as domain-specific translation behaviour, and correspondingly the extent to which domain adaptation techniques can be used to mitigate gender bias effects in NMT.

1.2 Contributions

We describe the specific contributions of this thesis, many of which have previously been described in reviewed publications:

- We show that simple, data-based domain adaptation to new domains can be very effective in scenarios where the domain is known and the adaptation data closely related to the general and test domain (Saunders, Stahlberg, and Byrne, 2019).
- However, we demonstrate that adaptation when the test domain is unknown can lead to imperfect adaptation or catastrophic forgetting, which we propose addressing with regularized training (Saunders, Stahlberg, de Gispert, et al., 2019; Stahlberg, Saunders, de Gispert, et al., 2019).
- More subtly, domain mismatch between adaptation and test data can lead to exposure bias effects. We develop a more robust form of Minimum Risk Training (MRT), which reduces these effects (Saunders and Byrne, 2020a; Saunders, Stahlberg, and Byrne, 2020).
- We extend Bayesian Interpolation with source information and apply it to NMT decoding with models on multiple individual domains, adaptively weighting ensembles without relying on test domain labels. Our scheme out-performs the 'oracle' case where the test domain is known (Saunders, Stahlberg, de Gispert, et al., 2019), and can be smoothed for additional robustness (Saunders, Stahlberg, and Byrne, 2019).
- We demonstrate that using different source representations of the same data at training and inference time can give improvements for both higher- and lower-resource domains when translating between linguistically distant languages (Saunders, Feely, et al., 2020).
- We show that models trained on different target language representations of the same data can have complementary attributes reminiscent of models trained on different domains, and develop a method for combining such models in an ensemble (Saunders, Stahlberg, de Gispert, et al., 2018; Stahlberg, Saunders, Iglesias, et al., 2018).
- We frame the task of countering gender bias effects in NMT as a domain adaptation problem, and apply various domain adaptation techniques to improve the situation (Saunders and Byrne, 2020b; Tomalin et al., 2021).

- We highlight some particular difficulties in NMT relating to gender bias, in that even apparently successful bias-reduced NMT systems over-generalize from available information. We propose introducing word-level tags during fine-tuning to improve matters (Saunders, Sallis, et al., 2020).

1.3 Structure of the thesis

This thesis begins with a review of relevant prior work. Chapter 2 introduces NMT with a summary of popular approaches in four specific categories: translation data, model architecture, training procedure and inference procedure. Chapter 3 describes what is meant by a domain in this thesis, then provides a thorough overview of approaches to domain adaptation for machine translation in the same four categories explored in Chapter 2. The literature review concludes with a summary of the gender bias problem in NMT, which we frame as a domain adaptation problem.

The remainder of the thesis presents original work. Chapter 4 includes some initial experiments on data selection and curricula for translation domains. We emphasize the advantages of data-centric approaches to NMT domains, as well as demonstrating some inherent disadvantages, such as exposure bias and over-fitting to a narrow domain, which are addressed in the following two chapters. Chapter 5 presents our investigations into parameter adaptation schemes for NMT fine-tuning, applying regularization techniques to address over-fitting and catastrophic forgetting, and developing a robust discriminative training scheme that can address exposure bias. Chapter 6 presents our work exploring multi-domain ensembling, combining multiple different-domain models in a domain-adaptive manner at inference time.

We conclude with two case studies on our own work. Both highlight the potential benefits of applying multi-domain adaptation techniques to aspects of NMT not typically considered to be domains. In Chapter 7 we consider different data representations for translation between English and more linguistically distant languages, and show that considerations typically made for domain adaptation can be useful, especially at inference time. In Chapter 8 we frame the task of mitigating the effects of gender bias in NMT as a domain adaptation problem, applying data, adaptation and inference techniques from throughout the rest of the thesis to address the problem. Conclusions are given in Chapter 9.

Much of the original work in this thesis has previously been published in reviewed conference or workshop proceedings: a list of relevant publications is given in Appendix A.

Chapter 2

Neural machine translation: a review

In this chapter we review recent developments in Neural Machine Translation (NMT). While research in the field has expanded hugely in recent years, we focus particularly on approaches which lay the groundwork for experiments in this thesis.

Machine Translation (MT) seeks to automatically translate written text between natural languages, from a source language to a target language. Originally accomplished with statistical MT (SMT) techniques consisting of word- and phrase-based frequency models, state-of-the-art performance in high-resource language pairs like English-German has been achieved by NMT since 2016 (Bojar, Chatterjee, et al., 2016). More recently Sennrich and Zhang (2019) have shown that with appropriate settings, NMT can outperform SMT on low-resource language pairs as well.

We structure this review chapter to correspond to the process of developing and applying an NMT model. First, training sentences in source and target languages must be represented as sequences of tokens in a fixed vocabulary (Sec. 2.1). An NMT model architecture must also be determined (Sec. 2.2). Training can then take place, where the model parameters are adjusted according to a training objective and training examples (Sec. 2.3). Finally, inference can be performed using a trained NMT model to translate previously unseen sentences (Sec. 2.4).

2.1 Representing language for NMT

Typically NMT models take as a single training example a sequence of tokens $\mathbf{x}$ representing a source language sentence and a sequence of tokens $\mathbf{y}$ representing the corresponding target language sentence.

These tokens are presented to the neural network as integer IDs $\in V$ where V is the corresponding source or target vocabulary. Depending on modelling choices, a token may

represent a word, a subword unit, or a character. Alternatively it may represent some other feature of the sentence, such as an element of its syntactic parse tree.

The language representation must be chosen with two goals in mind:

- Conveying information about the sentence pair that is useful for translation.
- Staying within practical constraints on the size and computation available for training and storing the NMT model.

The first goal could be met by representing every unit of the surface text in a unique way. Unknown or out-of-vocabulary (OOV) words would detract from this.

The second goal may be met by reducing the computational demand of the text representation. Extremely large vocabulary sizes or very long sentence representations would detract from this.

These goals are often at odds. For example, an extremely large word vocabulary might have few or no OOV terms, but involve large computational demands. In this section we explore approaches taken to balance these goals when representing language for neural translation models.

2.1.1 Word vocabularies

An NMT network vocabulary is typically limited to tens of thousands of tokens. This is primarily due to the computational complexity of the softmax which is used to map word embeddings to discrete tokens (Sec. 2.2). Early approaches to NMT determine source and target language vocabularies by identifying all unique words in the respective training sets and ordering them by occurrence frequency. The model source vocabulary is then typically the top $|V|$ source words by frequency (Cho, van Merriënboer, Gulcehre, et al., 2014) or all words appearing more frequently than some pre-determined cut-off (Kalchbrenner and Blunsom, 2013), and likewise for a separate target vocabulary. Any out-of-vocabulary (OOV) words are typically represented by a special `UNK` token.

Extremely large word vocabularies are possible under such approaches using a hierarchical approximation to the expensive softmax operation (Jean, Cho, et al., 2015). Nevertheless, OOV words are inevitable. New domains may use vocabulary very distinct from training data, existing words may be differently inflected or compounded, and new words will be conceived (Kornai, 2002). Word-based NMT models must therefore have strategies for encountering OOVs.

A translation containing `UNK` tokens can immediately be improved by replacing these tokens with correct target language words. Consequently, word-based NMT relies heavily on

rare word replacement techniques (Jean, Firat, et al., 2015; Li, Zhang, et al., 2016; Luong, Sutskever, et al., 2015). Rare word replacement involves first identifying the source word corresponding to the output `UNK` with some form of alignment model or labelling system, and next replacing the `UNK` with a better translation. This translation could be from an external lexicon, or could simply be the source word itself - particularly useful for named entity translation in which source words may be copied directly to the target.

2.1.2 Subword vocabularies

Multi-character subword vocabularies

A drawback of word vocabularies is their inherent treatment of all words as distinct, unrelated entities. In reality this is not always the case. For example, morphologically rich languages like Hebrew may have many words which are inflections of some root word, and agglutinative languages like Turkish contain compounds formed of many other words.

Another weakness of word-level vocabularies is sparsity. Zipf's law states that the occurrence probability of a word is inversely related to its vocabulary rank (Zipf, 1949), meaning a large proportion of vocabulary words in a language occur very rarely in a given corpus. Representing each of these rare words as a unique vocabulary item is therefore inefficient. It also may not allow good learned representations, since individual vocabulary items are seen infrequently during training.

Subword vocabularies address these weaknesses. They allow compound or inflected words to share component subwords, conveying their relatedness. They also represent individually rare terms as sequences of shorter segments, allowing a denser and more efficient vocabulary representation.

Sennrich, Haddow, and Birch (2016d) first propose NMT on sequences of multi-character subword units obtained using the Byte Pair Encoding (BPE) algorithm (Gage, 1994). A BPE vocabulary is initialized with every character appearing in the training data, and all words represented as character sequences. Word endings are marked with a special character, or all words may be separated with a special character as in the WordPiece variation proposed by Wu, Schuster, et al. (2016). The BPE algorithm proceeds by counting all token pairs and iteratively merging the most frequent pair to produce a new token. Very frequent words eventually become single tokens, as they would in a word-based vocabulary. Unseen words can in the worst case be represented as character sequences. BPE vocabularies can be learned separately for the source and target language, or a joint vocabulary can be learned on both languages together. The latter is useful for translating between related languages which are likely to share cognates, such as the English-German language pair.

Alternative schemes for multi-character subword vocabularies have been proposed, based on syllables (Assylbekov et al., 2017), language model scores (Kudo, 2018) or linguistically-informed word segmentation (Ataman et al., 2017; Huck, Riess, et al., 2017; Macháček et al., 2018). However, frequency-based BPE decomposition has become broadly accepted as an effective default vocabulary scheme for NMT.

More recent work on subword vocabularies has attempted to improve NMT robustness by optimizing BPE granularity. Ding et al. (2019) find that smaller numbers of subword merges give better performance for low-resource language pairs, as in a low-resource domain segmentation must be more aggressive for individual subwords to remain frequent. Gallé (2019) and Salesky et al. (2020) similarly find that BPE is most effective when giving set coverage with high-frequency subwords while keeping the overall sequence lengths short.

We note that BPE only allows truly open vocabulary translation if all possible characters are represented in the training data, including the many tens of thousands of characters that are representable by the Unicode standard (Needleman, 2000). In practice very high character coverage can usually be achieved for individual languages, especially using character normalization (Kudo and Richardson, 2018). A recently proposed byte-level subword scheme has the ability to represent any Unicode character, and has been shown to perform comparably to regular BPE while allowing complete vocabulary sharing across languages for multilingual models (Wang, Cho, et al., 2020).

Character vocabularies

An extremely aggressive subword segmentation scheme represents words and sentences as sequences of individual characters. A character vocabulary with full character coverage can represent any unknown word, as with BPE. However, the character vocabulary has the advantage of being smaller than a BPE vocabulary. Early attempts at character-based NMT involved additional network elements to combine information from characters in the same word (Costa-jussà and Fonollosa, 2016; Johansen et al., 2016; Kim, Jernite, et al., 2016; Ling et al., 2015). Luong and Manning (2016) use character-based networks only for OOV words, outperforming contemporary OOV-replacement strategies (Jean, Firat, et al., 2015)

Later approaches explore segmentation-free character NMT (Chung et al., 2016; Lee, Cho, et al., 2017) or byte NMT, which is similarly motivated but extensible to multilingual translation (Costa-jussà, Escolano, et al., 2017). This approach is particularly beneficial for languages with no inherent word boundaries, like written Chinese. Such languages must be segmented before word- or subword-based NMT. For languages with word segmentation NMT quality is still sensitive to punctuation tokenization (Domingo et al., 2018), which is less relevant for character-based NMT.

A drawback of character-based sentence representations is their length. This sentence contains only fourteen word tokens but nearly six times as many characters. Increasing sequence length increases computational requirements during training, and exponentially increases the space of possible translations to be explored during decoding. Deeper models or compression mechanisms can mitigate the quality effect, but addressing the computational cost remains challenging (Cherry et al., 2018).

Another disadvantage of character-based NMT is that characters representing syllables (e.g. Japanese kana) or phonemes (e.g. the Roman alphabet) do not usually correspond to words or morphemes. Exceptions include grammatical particles or other single-character words, but the number of exceptions is necessarily limited by the alphabet's size, normally tens of characters. Character-based NMT may therefore not enable efficient translation of meaning for languages with such alphabets.

Sub-character vocabularies

Alphabetic or syllabic alphabets are typically small and their characters do not usually convey semantic meaning. By contrast logographic alphabets like Chinese may have tens of thousands of logograms: characters representing one or more words, morphemes or concepts as well as conveying pronunciation. Treating logograms as units of meaning is therefore reasonable. However, logograms are extremely sparse: the Chinese character frequency distribution falls below that predicted by Zipf's law (Shtrikman, 1994). Inevitably some logograms will be rare or not present in the training data.

Many logograms share sub-character components called radicals. Early Chinese dictionaries identify just 214 radicals, each carrying semantic meaning, where the most frequent 10 radicals account for over ten thousand traditional Chinese characters (Mei, 1615; Yushu, Tingjing, et al., 1716). An intuitive approach to the logogram sparsity problem in natural language processing (NLP) uses the radical decomposition in the task. This has been shown to help incorporate a semantic component into character embeddings (Sun, Lin, et al., 2014) and to improve language modelling (Nguyen, Brooke, et al., 2017). Sub-character decomposition has also been applied to sentiment classification (Ke and Hagiwara, 2017), with mixed results (Karpinska et al., 2018).

Sub-character work in NMT has focused on use of shared radicals to improve Chinese-Japanese NMT (Zhang and Komachi, 2019; Zhang and Komachi, 2018). Such work typically uses radicals as analogy with characters in an alphabetic writing systems, decomposing all logograms and learning BPE vocabularies over radicals instead of over the logograms. This pre-supposes that all logograms benefit from decomposition. In Sec. 7.2 we demonstrate

that this is not necessarily the case for more distant language pairs, and discuss our own approaches to the problem of logographic character coverage (Saunders, Feely, et al., 2020).

2.1.3 Syntactic representations and tags

The language representations discussed so far have represented sentences simply in terms of language granularity: words, subwords, characters and sub-characters. A separate approach is augmenting source or target sequences with elements not present in the surface representation of the original sentences. Common examples are syntactic annotations or externally-defined tags.

Various forms of syntactic annotation have been incorporated into NMT: inflection agreement models (Green and DeNero, 2012), Combinatory Categorial Grammar (CCG) tags (Nadejde et al., 2017; Zhang and Clark, 2011), Part-of-Speech (POS) tags and syntactic dependency labels (Sennrich and Haddow, 2016), lemmas with morphological features (Tamchyna et al., 2017) and linearized constituency parses (Aharoni and Goldberg, 2017; Currey and Heafield, 2019). In Sec. 7.3 we explore some of these schemes and suggest some practical considerations for training and conducting inference with such representations.

Another way to augment a source or target sequence is with tags, which can be used to indicate a particular feature of the sentence. Previous work has used tags to convey the domain of a sentence (Kobus et al., 2017), the gender of a speaker (Vanmassenhove et al., 2018), target language formality (Feely et al., 2019; Sennrich, Haddow, and Birch, 2016a) and whether to translate a word using custom terminology (Dinu et al., 2019). Such work typically incorporates tags into all NMT data from the start of training, implicitly requiring the availability of reliable tags for all training data. In Sec. 8.5, by contrast, we explore the introduction of tags during adaptation for fine-grained control of translated gender inflections.

2.1.4 Representing document context

There has been much recent interest in including context in NMT language representations outside of the source or target sequence, such as the previous source sentence (Tiedemann, Scherrer, et al., 2017) or even providing whole document context (Junczys-Dowmunt, 2019; Macé and Servan, 2019). Early attempts to incorporate the previous source sentence into recurrent NMT showed potential improvements in terms of BLEU score (Wang, Tu, et al., 2017) or cross-lingual pronoun prediction (Jean, Lauly, et al., 2017). However, more recent investigations into document-level NMT with stronger self-attention baselines has given results that are as yet inconsistent (Stahlberg, Saunders, de Gispert, et al., 2019), although

they may be applicable in scenarios such as lexical cohesion or anaphora resolution (Voita, Sennrich, et al., 2019; Voita, Serdyukov, et al., 2018).

Kim, Tran, et al. (2019) demonstrate that improvements from document-level NMT may not even be interpretable as use of extra-sentence context, but that the extra context may simply provide a regularization effect. Moreover, they show that context-specific information, such as topic for lexical choice, can be retained in very minimal forms – effectively as individual tags – rather than necessitating encoding an entire additional sentence or document. For these reasons we focus on sentence-level translation in this thesis, although in Sec. 5.3 we explore ways to incorporate mini-batch level 'context', whether from a real document or not, into NMT training objectives.

2.2 Neural translation model architecture

Section 2.1 discussed various ways sentences can be represented for NMT, whether as sequences of words, subwords, or characters, or as linearized parse trees. In this section we treat all source sequences **x** and target sequences **y** as simply sequences of integer values.

The NMT model input is a sequence of these integers:

$$\mathbf{x} = x_1, ..., x_I, x_i \in V_{src} \tag{2.1}$$

And the NMT model must produce a sequence of integers:

$$\mathbf{y} = y_1, ..., y_J, y_j \in V_{trg} \tag{2.2}$$

In this thesis we follow the terminology of Cho, van Merriënboer, Gulcehre, et al. (2014) in referring to the subnetwork which maps *from* **x** as an encoder, and to the subnetwork which maps *to* **y** as a decoder. We refer to the phase of updating neural network parameters given **x** and **y** as training, and the phase where parameters are fixed and new hypotheses are generated without reference tokens **y** as inference. We refer to the more general process of producing a sequence with the decoder as decoding whether or not **y** is available.

We therefore seek a neural network that can learn a mapping between **x** and **y** which generalizes to unseen **x** during inference. Many neural network architectures are capable of this. In this section we discuss the development of architectures that have become widely used for NMT in recent years.

2.2.1 Continuous word embeddings

The first stage in an NMT model is mapping a sequence of integers $\mathbf{x} \in V_{src}$ to a continuous representation of each integer known as a word embedding. Vocabulary sizes $|V|$ for standard NMT models are typically $>$ 10K. A $|V|$-dimension 'one-hot' representation of a single vocabulary item is too large for neural network operations to be tractable on potentially thousands of words per mini-batch of sentences. Moreover, one-hot representations are discrete: the effect of changing a single word does not necessarily generalize to the effect of changing other words.

For these reasons Bengio, Ducharme, et al. (2003) proposed associating each vocabulary word with a continuous word feature vector, also known as a word embedding. These are a more tractable size: d where typically $d \leq 1024$. An embedding matrix $W \in \mathbb{R}$ maps from $|V|$ discrete dimensions to d continuous dimensions. Parameters of W are learned by error back-propagation.

Word embeddings can be learned using various context-based objectives. For example, Bengio, Ducharme, et al. (2003) map a sequence of word embeddings to a conditional probability distribution for the next word over words in V, and Collobert and Weston (2008) use the context of a word to predict the word itself. These formed early neural language models (LMs) which could be used as translation models (Le et al., 2012) and language models (Schwenk, 2012) for phrase-based SMT systems.

The dimension d embeddings trained for such objectives may exhibit local smoothness and therefore generalize: that is, words with similar sequence contexts will tend to have similar continuous representations (Collobert and Weston, 2008; Turian et al., 2010). Later work shows that word embeddings trained with context-related objectives can encode semantically meaningful results. Words have similar embeddings if they belong to the same category, for example if both are countries or colours (Collobert, Weston, et al., 2011; Mikolov, Sutskever, et al., 2013). The vector difference between pairs of word embeddings may also encode relationships between those words (Mikolov, Corrado, et al., 2013).

2.2.2 Sequence encoders

Neural networks that learn word embeddings from context as described in Sec. 2.2.1 must involve a sequence encoder. A sequence encoder is a subnetwork which encodes information about an input sequence of tokens, as opposed to a single input token. For example, neural LMs as described above effectively encode a sequence of input word embeddings as a sequence embedding. The probability distribution for the next output word is then conditioned on this sequence embedding.

Early work used feed-forward neural networks to learn sequence embeddings for language processing tasks. However, feed-forward models are inherently limited in the length of sequence they encode. For a feed-forward LM with encoding dimension n, word probabilities are conditioned on an n-gram sequence embedding, rather than whole-sentence context. A sequence embedding conditioned on all available context is more useful, particularly for translation, where the correct output can be affected by long-range dependencies between source words.

Convolutional and recurrent sequence encoders

Sequence embeddings which do not require a fixed-length input sequence can be encoded using Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs.) A CNN applies a sliding filter window to a variable-length sequence of word embeddings, generating features which can be combined hierarchically to produce a fixed-length sequence embedding. However, the convolutional approach alone typically loses word-ordering information unless it is explicitly reintroduced (Kalchbrenner and Blunsom, 2013; Kalchbrenner, Grefenstette, et al., 2014; Socher et al., 2011).

RNNs have been more frequently used to encode sequences for NLP (Mikolov, Karafiát, et al., 2010; Sutskever, Martens, et al., 2011; Sutskever, Vinyals, et al., 2014). A standard RNN maintains a hidden state $\mathbf{h}$, which may be initialized as a zero vector or by random sampling (Zaremba et al., 2014). At each step t through input sequence $\mathbf{x}$, $\mathbf{h}$ is updated:

$$\mathbf{h_t} = f(\mathbf{W}^{|\mathbf{h}||\mathbf{h}|}\mathbf{h_{t-1}} + \mathbf{W}^{|\mathbf{h}||\mathbf{x}|}x_t), \text{where } f \in \{\text{sigm}, \tanh\} \tag{2.3}$$

Here $\mathbf{W^{ab}}$ are parameter matrices mapping from b dimensions to a dimensions which are learned using gradient back-propagation through time (Rumelhart et al., 1986; Werbos, 1988). We use $|\mathbf{h}|$ as notation for the size of embedding $\mathbf{h}$. The end of the sequence is marked by a special end-of-sentence (EOS) token. After the whole sequence of T inputs is seen $\mathbf{h_T}$ embeds the whole of $\mathbf{x}$, since $\mathbf{h_T}$ has a recurrent dependence on all previous inputs.

Error gradients must be backpropagated through each step of the input to train the RNN. This causes problems as T becomes large. Repeated products of very large long-distance gradient contributions may grow exponentially larger than local gradient contributions. Conversely, repeated products of small long-distance gradient contributions may vanish (Bengioy et al., 1994). Weight regularization or gradient clipping can mitigate the exploding gradient problem (Mikolov, 2012; Pascanu et al., 2013). Long short-term memory cells (LSTMs, Hochreiter and Schmidhuber (1997)) and gated recurrent units (GRUs, Cho, van Merriënboer, Gulcehre, et al. (2014)) have been introduced to avoid the vanishing gradient

problem as RNN variants that 'adaptively remember and forget' with reset and update operations.

Reverse and bidirectional sequence encoders

The state of an LSTM is in principle affected by past inputs across arbitrarily long-distance 'time lags' from the current input. However, experiments by Sutskever, Vinyals, et al. (2014) suggest the RNN state is most strongly affected by the most recent inputs. They propose the simple alternative of embedding the source sentence right-to-left, starting with the final word, which they find improves NMT results.

A related problem is that RNNs can only model arbitrarily long sequential relationships in one direction. One proposed solution is the Bidirectional RNN (BiRNN, Schuster and Paliwal (1997)), which has had particular impact on NMT (Bahdanau, Cho, et al., 2015). A BiRNN reads the source sentence in two passes of Eq. 2.3: a forward pass from first to last token and a backwards pass from last to first, producing two sets of encodings. The overall state $\mathbf{h_t}$ at source sequence position t is found by concatenating the corresponding states from both passes. This state then has a strong dependence on the tokens immediately preceding and following x_t.

2.2.3 Sequence decoders

A sequence decoder is a sub-network which generates a sequence of discrete tokens conditioned on a sequence embedding (Sec. 2.2.2). In NMT, the generated output is a translation hypothesis that is conditioned on an input sentence representation. The decoder may also be conditioned on an output sequence representation. During training this is usually the reference sentence. During inference the reference is not available, so the decoder is conditioned on its own hypothesis translation.

The decoder output is fed through a softmax which produces a probability distribution over tokens in V_{trg}. During inference the discrete model output $\hat{y_j}$ can then be chosen according to an inference algorithm. A common and fast choice is the greedy approach, which selects the most probable output token given the distribution. Different approaches are discussed in Sec. 2.4.

Recurrent decoders

An early NMT sequence decoder proposed by Sutskever, Vinyals, et al. (2014) consists of an LSTM RNN with hidden state s. Input y_{j-1} consists of a reference token during training, and the previously generated token $\hat{y}_{j-1}$ during inference. Initial input y_0 is a special

beginning-of-sentence (BOS) token. Initial state $\mathbf{s}_0$ is set to the source sequence embedding $\mathbf{h_T}$ found by the encoder in Eq. 2.3. A single decoding step takes place as follows:

$$\mathbf{s_j} = f(\mathbf{U}^{|\mathbf{h}||\mathbf{h}|}\mathbf{s_{j-1}} + \mathbf{U}^{|\mathbf{h}||\mathbf{y}|}y_{j-1}), \text{where } f \in \{\text{sigm}, \tanh\} \tag{2.4}$$

$$p(\hat{y}_j|x, y_{1:j-1}) = \text{softmax}(\mathbf{U}^{|\mathbf{y}||\mathbf{h}|}\mathbf{s_j}) \tag{2.5}$$

Decoding terminates when the EOS symbol is produced. As in Eq. 2.3 $\mathbf{U}^{\mathbf{ab}}$ are parameter matrices mapping from b dimensions to a dimensions. If the source and target vocabularies are shared, these parameters may be the same as the corresponding $\mathbf{W}^{\mathbf{ab}}$ matrices from the encoder.

Cho, van Merriënboer, Gulcehre, et al. (2014) propose a similar RNN decoder based on GRUs. Their hidden state update is additionally a function of a context vector $\mathbf{c}$, which they set as the source sequence embedding $\mathbf{h_T}$. This allows conditioning on the input sentence directly instead of via the decoder's recurrence:

$$\mathbf{s_j} = f(\mathbf{s_{j-1}}, y_{j-1}, \mathbf{c}) \tag{2.6}$$

They also propose a size m maxout pooling operation in the computation of the output from the decoder hidden state (Goodfellow et al., 2013):

$$p(\hat{y}_j|x, y_{1:j-1}) = \text{softmax}(\mathbf{U}^{|\mathbf{y}|\mathbf{m}}\max(\mathbf{U}^{\mathbf{m}|\mathbf{h}|}\mathbf{s_j})) \tag{2.7}$$

Attention-based recurrent decoders

The decoder of Eq. 2.6 maintains a direct dependence on the initial sequence embedding via $\mathbf{c}$, but must still encode all information from a variable-length source sentence into a single fixed-size embedding. Cho, van Merriënboer, Bahdanau, et al. (2014) observe that, for a straightforward RNN-based encoder-decoder network, translation quality decreases as source sentence length increases. As discussed in Sec. 2.2.2, sequence embeddings from t steps through a recurrent encoder are strongly affected by inputs from locations near t and tend to lose contributions from distant inputs. They may therefore fail to adequately represent the whole source sentence.

An alternative approach is an attention-based decoder, which to the best of our knowledge was first applied to NMT by Bahdanau, Cho, et al. (2015). This approach uses the decoder architecture described in Eqs. 2.6 and 2.7 but calculates the context vector $\mathbf{c_j}$ for each decoder time-step j via an attention mechanism. Using the later formalism of Vaswani, Shazeer, et al. (2017), given a query Q and a mapping of key-value pairs K and V, an attention mechanism returns V weighted by some 'score', which is a similarity function between Q and K.

$$c_j = \text{softmax}(\text{score}(Q,K))V \tag{2.8}$$

Bahdanau, Cho, et al. (2015) store all forward-backward states of a BiRNN source sentence encoding, $\{\mathbf{h_1},...,\mathbf{h_T}\}$, and set V as a $T \times |\mathbf{h}|$ matrix where $V_t = h_t$. They further define $K = V$ and $Q = s_{j-1}$, and use a feed-forward network as the score(.) function. Because h_t is most strongly affected by source token x_t, the attention mechanism weights $\text{softmax}(\text{score}(Q,K)) \in [0,1]$ act as a soft alignment model, indicating the importance of source token x_t for producing c_j and consequently for producing target output y_j.

Luong, Pham, et al. (2015) explore variations on this attention mechanism. They set $Q = s_j$ – applying alignment after the decoder state update – and experiment with 'local' attention, in which V_t is down-scaled if t is far from j. They also use a dot-product score function:

$$\text{score}(Q,K) = QK^T \tag{2.9}$$

This can be implemented more efficiently than the feed-forward network of Bahdanau, Cho, et al. (2015) as the dot product relies only on matrix multiplication, not an additional parameterized network. Both Bahdanau, Cho, et al. (2015) and Luong, Pham, et al. (2015) find that attention mechanisms improve translation performance on long sentences in particular.

2.2.4 Purely attention-based encoder-decoder networks

Recurrent encoder-decoder models necessarily involve sequential calculation for a given training example. Such calculations cannot be parallelized, introducing a bottleneck on training time as sequence lengths increase. To improve NMT parallelizability Vaswani, Shazeer, et al. (2017) propose the Transformer model, a purely attention-based architecture without recurrence.

The Transformer network achieved strong improvements in parallelizability and performance over RNN-based models, and remains state-of-the-art for NMT at the time of writing. Experiments in the remainder of this thesis therefore use the Transformer model architecture as described here, unless stated otherwise.

Transformer attention and self-attention

The Transformer uses encoder-decoder attention as in Sec. 2.2.3 to relate K and V from the encoder output to Q from the decoder state. However, where recurrent models determine source and target sequence representations with sequential calculations, the Transformer

model relies purely on self-attention for these representations. This improves speed and parallelizability.

The attention mechanism described in Sec. 2.2.3 related different positions in different sequences in order to learn a 'context representation'. Self-attention instead relates different positions in a *single* sequence in order to calculate that sequence's representation. A self-attention embedding is determined as in Eq. 2.8, where Q, K and V are all projected from the same sequence. That sequence consists of source embeddings x at the input to the encoder, target embeddings y at the input to the decoder, or the output of the previous layer in a multi-layer encoder / decoder (Sec. 2.2.5).

Use of self-attention has some practical implications. In the decoder, the output of the self-attention score function is masked to $-\infty$ before the softmax for any positions $> j$ to prevent the decoder from attending to future target tokens. The Transformer model uses a scaled dot-product attention score function: this is the dot-product score of Eq. 2.9 scaled down by the embedding dimensionality, $|\mathbf{h}|^{\frac{1}{2}}$. Finally, self-attention does not convey the absolute position of a token in a sequence as recurrent encodings do. Vaswani, Shazeer, et al. (2017) reintroduce this information using positional embeddings. These are functions of token position which can be added to the token embeddings directly. Positional embeddings can be determined through pre-defined functions or learned jointly with the rest of the network.

Multi-head attention

So far, attention mechanisms have been described in terms of a single attention function with the same dimensionality $|\mathbf{h}|$ as Q, K and V. For the Transformer model, Vaswani, Shazeer, et al. (2017) instead propose multi-head attention. Multi-head attention projects the attention on each input H times with H different attention 'heads', each attending to a different Q, K and V of dimensionality $\frac{|\mathbf{h}|}{H}$. The attention head outputs are concatenated into a dimensionality $|\mathbf{h}|$ multi-head attention embedding which can represent joint attention on different positions in the same sequence.

2.2.5 Multi-layer networks

NMT models generally follow an encoder-decoder architecture. Each encoder or decoder discussed so far – recurrent, convolutional, self-attention-based – may be implemented as a multi-layer network. For example, the Transformer as described in Vaswani, Shazeer, et al. (2017) has an encoder and decoder each composed of a stack of 6 identical layers. Each layer carries out its operation (e.g. self-attention) on the output of the layer before. The final layer

in the encoder is used as input to the decoder, and the final layer in the decoder is used to produce the output.

In principle multi-layer networks are capable of learning more fine-grained language representations than single-layer networks, simply because they have more parameters. In practice multi-layer networks are susceptible to training difficulties since objective function gradients must be propagated through more layers.

A common way to improve gradient propagation is adding residual networks around each layer – that is, the output of a layer $f(z)$ becomes $f(z)+z$ (He et al., 2016). Each layer in the encoder or decoder subnetwork then has access to the original subnetwork input. Residual connections have been found to be necessary when training deep recurrent models (Britz, Goldie, et al., 2017) and Transformer models (Chen, Firat, et al., 2018).

A related challenge is that the gradients of weights in a layer may become strongly correlated with the gradients of a previous layer. Known as Internal Covariate Shift (ICS), this correlation makes convergence challenging as training proceeds, since parameter change towards convergence in one layer may disturb parameters away from convergence in subsequent layers (Ioffe and Szegedy, 2015). Techniques proposed to avoid ICS include batch normalization (Ioffe and Szegedy, 2015) and layer normalization (Ba et al., 2016). While there is some debate over whether these techniques really address ICS (Santurkar et al., 2018), layer normalization is used around layers in Transformer models (Vaswani, Shazeer, et al., 2017) and has found to be necessary for training large Transformer models (Chen, Firat, et al., 2018).

When such architectural 'tricks' are applied, deep models have been shown to outperform equivalent shallower models for NMT in some settings (Wang, Li, et al., 2019). These techniques effectively move the bottleneck on model size towards constraints such as memory footprint and training time. However, we do not consider model depth a panacea, and note that recent work has in some cases found better performance for shallower models, especially for low-resource translation (Nguyen and Chiang, 2018; Sennrich and Zhang, 2019).

2.3 Training NMT models

Once an NMT model architecture has been determined as in Sec. 2.2, its parameters must be adjusted so as to produce a mapping between source sequences $\mathbf{x}$ and target sequences $\mathbf{y}$. NMT model parameters are trained by backpropagation (Rumelhart et al., 1986), typically using some form of Stochastic Gradient Descent (SGD) optimizer. These gradients must be determined for some objective on the training data.

Standard training objectives, such as cross-entropy loss, use both the source sentence and reference sentence during training. However, during inference only the source sentence and the prefix of the model's own hypothesis $\hat{\mathbf{y}}$ are available. An auto-regressive sequence decoder therefore experiences a discrepancy between conditioning during training and inference, commonly known as exposure bias (Bengio, Vinyals, et al., 2015; Ranzato et al., 2016). The need to improve performance while avoiding exposure bias has motivated alternative objectives and regularization methods.

In this section we explore several objective and regularization functions that have proven popular for training NMT models, as well as other factors that influence training such as choice of optimization algorithm and minibatch size.

2.3.1 Objective functions

Cross-entropy loss

Since early development of neural networks trainable by gradient descent (Rumelhart et al., 1986), it has been proposed that a generalizeable approach to neural network training is varying weights θ in the gradient direction of the log likelihood in order to maximize the log likelihood of training examples (Baum and Wilczek, 1988; Levin and Fleisher, 1988). This is known as Maximum Likelihood Estimation (MLE).

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \log P(\mathbf{y}|\mathbf{x};\theta) \tag{2.10}$$

MLE is equivalent to minimizing the cross-entropy loss L_{CE} between the generated output distribution and the target sequences where there is a single ground-truth label $q(y' = y_j|\mathbf{x};\theta) = \delta(y_j)$ for each token:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{j=1}^{|y|} -\log P(y_j|\mathbf{y}_{1:j-1},\mathbf{x};\theta) = \underset{\theta}{\operatorname{argmin}} L_{CE}(\mathbf{x},\mathbf{y};\theta) \tag{2.11}$$

Continuing from this early work in effective neural network training, the majority of end-to-end NMT models minimize the cross-entropy loss as an objective function.

Minimum risk loss

As discussed in the opening of this section, MLE training is susceptible to exposure bias. MLE also experiences loss-evaluation metric mismatch, since it optimizes the log likelihood of training data while machine translation is usually evaluated with translation-specific metrics.

Discriminative training for MT was introduced for phrase-based SMT to bring model parameter learning objectives in line with evaluation metrics. This was achieved by minimizing the expected cost of model hypotheses in terms of a translation metric like document-level BLEU (Papineni et al., 2002), either by scoring across a set of sentences (Och, 2003) or by considering individual hypothesis contributions to the set (Watanabe et al., 2007).

Shen et al. (2016) extend these ideas to Minimum Risk Training (MRT) for NMT, using expected minimum risk at the sequence level with a sentence-level BLEU (sBLEU) cost for end-to-end NMT training. Given N sampled target sequences $\mathbf{y}_n^{(s)}$ and the corresponding reference sequences $\mathbf{y}^{(s)*}$ for the S sentence pairs in each minibatch, the MRT objective is:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{s=1}^{S} \sum_{n=1}^{N} \Delta(\mathbf{y}_n^{(s)}, \mathbf{y}^{(s)*}) \frac{P(\mathbf{y}_n^{(s)}|\mathbf{x}^{(s)};\theta)^{\alpha}}{\sum_{n'=1}^{N} P(\mathbf{y}_{n'}^{(s)}|\mathbf{x}^{(s)};\theta)^{\alpha}} \tag{2.12}$$

Hyperparameter α controls the smoothness of the sample probability distribution. Function $\Delta(.)$ measures hypothesis cost $\in [0,1]$, typically $1 - \text{sBLEU}(\mathbf{y}_n^{(s)}, \mathbf{y}^{(s)*})$, and model hypotheses are usually generated by auto-regressive sampling with temperature τ.

Edunov et al. (2018a) explore MRT with variations on these settings, using samples produced by beam search and calculating the sBLEU-based cost between pairs of samples. Wieting et al. (2019) use MRT for translation with sBLEU and with a sentence similarity metric. Outside of NMT, MRT has also been successfully applied to summarization (Ayana et al., 2016), speech recognition (Shannon, 2017), and machine translation post-editing (Tebbifakhr et al., 2018).

MRT is related to neural reinforcement learning (RL) which aims to directly optimize the evaluation measure at training time. We give a brief overview of RL approaches that have been applied to NMT. The actor-critic scheme (Sutton et al., 2000) optimizes over sequence rewards at each state of producing a hypothesis, and has therefore a natural application to recurrent NMT models (Bahdanau, Brakel, et al., 2017; Nguyen, Daumé III, et al., 2017). The Reinforce algorithm described by Williams (1992) aims to maximize a reward metric for samples from the model distribution and has been incorporated into NMT loss functions alongside the cross-entropy loss (Ranzato et al., 2016) or alone (Wu, Tian, et al., 2018; Wu, Zhao, et al., 2017). Reinforce-related approaches have also been applied to other areas of NLP that are commonly formulated as sequence-to-sequence problems, such as Grammatical Error Correction (GEC) (Sakaguchi et al., 2017).

MRT is of particular relevance to this thesis due to its property of robustness to exposure bias, which itself has been shown to be a particular problem if there is a domain mismatch between adaptation and test data (Müller et al., 2020). This feature of MRT is highlighted by Neubig (2016), who notes that MRT tends to produce sentences of the correct length without

needing length penalty decoding, and Wang and Sennrich (2020), who find MRT reduces exposure bias when the test data domain is very different from the training domain. Due to these recognized strengths of MRT, we explore it further in this thesis (Sec. 5.3).

2.3.2 Regularization

An alternative approach to exposure bias under MLE is to regularize MLE training[1]. The aim is to avoid over-fitting parameter weights to the training set. This can be achieved by adding a regularization penalty to the MLE loss function.

Output distribution regularization

Log-likelihood maximization as in Eq. 2.11 assumes that a ground-truth label is far more likely than all other labels. This objective encourages large differences in likelihood between training examples and language that does not appear during training. This can result in over-confidence and over-fitting to the training data, reducing the model's ability to cope with novel data during inference.

A solution to this problem is to incorporate some level of uncertainty into the distribution over output labels. Szegedy et al. (2016) propose 'label smoothing' (LS) for computer vision tasks using convolutional networks, which replaces the single target label $q(y'|\mathbf{x};\theta) = \delta(y_j)$ used to derive Eq. 2.11. Instead the label distribution is smoothed towards a uniform distribution over the target vocabulary by parameter ε:

$$q(y'|\mathbf{x};\theta) = (1-\varepsilon)\delta(y_j) + \frac{\varepsilon}{|V_{trg}|} \tag{2.13}$$

The cross-entropy objective function then becomes:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{j=1}^{|y|} \sum_{y' \in V_{trg}} -q(y'|\mathbf{x};\theta) \log P(y'|\mathbf{y}_{1:j-1}, \mathbf{x};\theta) \tag{2.14}$$

Label smoothing was made popular for NMT by its use in purely attention-based networks such as the Transformer model (Vaswani, Shazeer, et al., 2017), although it has also been shown to improve RNN-based NMT performance (Chen, Firat, et al., 2018).

Instead of smoothing the distribution over labels with a uniform distribution $\frac{1}{|V_{trg}|}$, the smoothing distribution can come from a teacher model, known as knowledge distillation (Hinton, Vinyals, et al., 2015). Label smoothing can also smooth towards a unigram dis-

[1] We consider this as a separately motivated task from regularization during domain adaptation of a pre-trained model, which is discussed in Sec. 3.4.2.

tribution over the vocabulary (Pereyra et al., 2017). These schemes effectively incorporate prior information about the target language distribution. Alternatively, Pereyra et al. (2017) suggest simply penalizing confident (i.e. low-entropy) output distributions as a regularization method for NMT among other tasks.

Objective function regularization

Rather than adjusting the ground truth output distribution before applying the loss function, a regularization term can be added to the loss function itself.

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}}[L_{CE}(\mathbf{x}, \mathbf{y}; \theta) + \lambda L_{\text{Reg}}(\theta)] \tag{2.15}$$

One simple case is for L_{Reg} is an L2 penalty term, $L_{\text{Reg}} = \sum_i \theta_i^2$. More translation-specific regularization terms can involve multi-task learning, in which the added loss term is an objective from another task. Proposed multi-task-related loss terms include a coverage term to address over- and under-translation (Tu et al., 2016), a right-to-left translation objective (Zhang, Wu, et al., 2019), the 'future cost' of a partial translation (Duan et al., 2020), or a target language modelling objective (Gülçehre et al., 2015; Sriram et al., 2018; Stahlberg, Cross, et al., 2018).

An alternative but related approach is dropout, which randomly omits a small subset of parameters θ_{dropout} from optimization for a training batch (Hinton, Srivastava, et al., 2012). This can be interpreted as regularization at a given training step with $L_{\text{Reg}}(\theta) = \infty$ for $\theta \in \theta_{\text{dropout}}$, 0 otherwise, but without the effect of numerical overflow.

2.3.3 Optimization choices

Optimizer

While Rumelhart et al. (1986) originally suggest learning parameter weights for neural networks by gradient descent, many variations have been proposed. Stochastic Gradient Descent (SGD) algorithms are sensitive to both minibatch size and learning rate. Training typically continues until loss function convergence, or until converged on some held-out validation set. Zeiler (2012) proposes AdaDelta optimization, which includes momentum terms calculated over recent gradients. The momentum terms serve to automatically reduce the learning rate near local minima, reducing parameter value oscillation. Adam optimization (Kingma and Ba, 2015) similarly uses momentum terms, determined as a running average of first and second gradient moments. Adam has been shown to empirically perform better than

Adadelta (Kingma and Ba, 2015) and is the optimizer algorithm used in this thesis unless otherwise stated.

Batch size

Data is fed into the neural network in minibatches, since computing the loss gradient over an entire dataset is generally impractical and computing loss gradient over individual examples is extremely slow. The mini-batch gradient may also be a less noisy estimate of the gradient over the entire training set than a gradient calculated on a single example. This effect of batch size on training performance can be interpreted in terms of signal-to-noise ratio of training example gradients (McCandlish et al., 2018). Using larger minibatches in training can therefore result in better final convergence, as demonstrated by Morishita, Oda, et al. (2017) and Neishi et al. (2017). Notably Smith et al. (2018) find increasing batch size during training requires fewer updates than the equivalent scheme of decaying the learning rate.

Batch size for NMT is often determined by number of tokens per mini-batch, rather than number of training sentence examples. In this case a model trained on longer sentences will on average see fewer training examples per minibatch than one trained on shorter sentences, and large batch size may be particularly important. We explore the importance of batch size in the context of varying text representation in Sec. 7.3.

2.4 Inference with NMT models

During training, the NMT model is provided with examples of source and target language sentences $\mathbf{x}$ and $\mathbf{y}$, and its parameters are adjusted in order to model $P(\mathbf{y}|\mathbf{x})$. During inference, the model has access only to $\mathbf{x}$, and must produce a target language translation hypothesis:

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}} P(\mathbf{y}|\mathbf{x}) \tag{2.16}$$

In standard autoregressive NMT inference, a single output token is produced at each inference step j. The model has access to the partial translation hypothesis, so this output token is additionally conditioned on all previously output tokens. In an ideal case:

$$\hat{y_j} = \underset{y_j}{\operatorname{argmax}} P(y_j|\mathbf{y}_{1:j-1}, \mathbf{x}) \tag{2.17}$$

Complete search through the $|V_{trg}|^j$ partial translations that are possible for the j^{th} output token is intractable. Stahlberg and Byrne (2019) show that exact search methods involving partial hypothesis likelihood pruning may still be impractically slow, and highlight that

such search procedures may not give good translations. Nevertheless approximations to this inference objective such as beam search perform well in practice.

In this section we review approaches to producing good translations during NMT inference which form the basis of experiments in this thesis.

2.4.1 Inference direction

While NMT inference as described in Eq. 2.17 takes place in a left-to-right (L2R) manner, it could equally be factorized right-to-left (R2L) (Liu, Utiyama, et al., 2016) - that is, producing the final token in the translation first. Other schemes for inference have been explored, such as middle-out generation (Welleck et al., 2019) and non-autoregressive generation (Gu, Bradbury, et al., 2018). In this thesis, we focus on the dominant approach of L2R inference.

2.4.2 Beam search

The most common algorithm for NMT inference is beam search. Beam search is a variant on best-first search which is well-established as a means of generating multiple speech and natural language sequence hypotheses generally (Greer et al., 1982) and of generating machine translation hypotheses specifically (Och and Weber, 1998).

Instead of an exact search through the hypothesis space, beam search tracks the top N partial hypotheses by log likelihood. At each inference step t, all possible single-token expansions of the beams are reranked, and the updated top N selected, until all beams terminate with an EOS marker or exceed a pre-determined maximum length. The special case where $N = 1$ is known as greedy search. In this case the model simply produces the most likely next token at each step.

Variations have included optimizing for cross-beam diversity (Li and Jurafsky, 2016; Vijayakumar et al., 2016) and discouraging the tendency of NMT models to produce short, inadequate translations. Koehn and Knowles (2017) highlight long sentence translation as a major challenge for NMT. Test sentences have the potential to be longer than any in the training set, which are normally length-filtered for more efficient training. During inference the negative effects of long sentences can be reduced by simply segmenting sentences before inference (Pouget-Abadie et al., 2014). However, there is evidence that inference schemes like beam search tend to produce short translations in most cases. Indeed, Stahlberg and Byrne (2019) find that extremely large beam sizes often assign the best score to the empty translation, indicating a failure to model adequacy. A typical solution is to apply length normalization during training (Murray and Chiang, 2018) or at inference time (Wu, Schuster, et al., 2016)

2.4.3 Ensembling

When performance is more important than decoding speed, a common approach is to perform inference with an ensemble of NMT models. This generally achieves better results than inference with one model alone (Dietterich, 2000; Frederking and Nirenburg, 1994; Hansen and Salamon, 1990).

In the context of MT, an ensemble of models allows consensus about the next tokens to expand and track in beams at each inference step (Rosti et al., 2007; Sim et al., 2007). Scores from translation models with different architectures (Stahlberg, de Gispert, and Byrne, 2018) or target side language models (Gülçehre et al., 2015; Vaswani, Zhao, et al., 2013) can be integrated. Many schemes for ensemble model combination exist, from simple majority vote to minimizing Bayes risk under a given evaluation metric. A thorough discussion of ensemble combination methods can be found in Rokach (2010).

In this thesis we focus on ensemble combination by weighting. That is, given K different models to ensemble and weights W_k defined for each model, the ensemble translates with

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}} P(\mathbf{y}|\mathbf{x}) = \underset{\mathbf{y}}{\operatorname{argmax}} \sum_{k=1}^{K} W_k P_k(\mathbf{y}|\mathbf{x}) \tag{2.18}$$

A downside of ensemble inference for NMT is the slow-down in the inference process, since the expensive softmax calculation must be carried out for each model individually. Ensembling also usually requires storing all ensemble models in memory simultaneously. Schemes such as ensemble knowledge distillation (Freitag, Al-Onaizan, and Sankaran, 2017; Fukuda et al., 2017) and ensemble unfolding (Stahlberg and Byrne, 2017) can enable a single model to reach similar performance as an ensemble of similarly trained models.

Each of these schemes for simplifying an ensemble of models into a single model assumes implicitly that all models in the ensemble produce the same form of output in the same way: for example, that all models produce subword sequences or syntactic sequences but not a mixture. They also assume that the same ensemble combination scheme is applicable in all cases: for example, that averaging scores from all ensemble models is a good combination scheme for all test sentences regardless of sentence or model domain. In Sec. 3.5.1 we discuss prior work on ensembles for domain adaptation where this is not the case. In terms of original work on the problem we describe our own approaches to domain adaptive ensembling in Chapter 6, and a scheme for multi-representation ensembles in Sec. 7.3.

Checkpoint averaging

If only a single model is trained, a simple alternative to model ensembling is possible by averaging parameters over individual checkpoints saved at different points in model training (Vaswani, Shazeer, et al., 2017). Popel and Bojar (2018) find that checkpoint averaging gives stronger performance than using individual checkpoints, with less performance fluctuation at a given point in training. Liu, Zhou, et al. (2018) investigate the optimal number of checkpoints to average for NMT, finding that too many checkpoints may result in decreasing scores.

2.4.4 Evaluating machine translation

The ultimate test of machine translation quality is human judgement. However, machine translation evaluation is often required in situations where manual human evaluation is not practical. When assessing incremental changes to many machine translation systems, or when optimizing a system directly with respect to an evaluation metric through MERT training (Och, 2003) or MBR decoding (Kumar and Byrne, 2004), hundreds or thousands of translation evaluations may be needed per model. In these cases human evaluations would be too time-consuming and expensive, so automatic evaluation metrics are used instead.

In general, automatic MT metrics compare generated translation hypotheses to human reference translations. A hypothesis that is similar to its reference is considered to be high quality, with the similarity measure varying between metrics. In this thesis we primarily use the BLEU metric (Papineni et al., 2002). BLEU is a function of n-gram precision p_n between hypothesis and human reference calculated over all hypotheses in a test set:

$$p_n = \frac{\sum_{\text{Hyps}} \sum_{\text{n-grams}} \min(\text{Count(Ref, n-gram)}, \text{Count(Hyp, n-gram)})}{\sum_{\text{Hyps}} \sum_{\text{n-grams}} \text{Count(Ref, n-gram)}} \tag{2.19}$$

The function Count(S, g) returns the number of occurrences of n-gram g in sequence S. Since hypotheses with high n-gram precision may still miss words that should appear in the reference, a brevity penalty (BP) is typically added:

$$BP = \begin{cases} 1, & |\text{Hyp}| > |\text{Ref}| \\ \exp\left(1 - \frac{|\text{Ref}|}{|\text{Hyp}|}\right) & \text{otherwise} \end{cases} \tag{2.20}$$

BLEU score is then the geometric average of n-gram precisions up to and including $n = N$, where usually $N = 4$. Interpolation weights w_n are normally set to $\frac{1}{N}$. The geometric average is scaled by the brevity penalty:

$$\text{BLEU} = BP \exp\Big(\sum_{n=1}^{N} w_n \log p_n\Big) \tag{2.21}$$

The maximum BLEU score is 1.0 for a translation hypothesis that perfectly matches the reference[2] and 0.0 for a translation containing no n-grams in the reference. Usually BLEU is scaled for reporting between 0.0 and 100.0.

BLEU is calculated at the corpus level, but it is sometimes necessary to evaluate sentence-level translation, for example for MRT optimization (Sec. 2.3.1). A typical approximation for this purpose is sentence-level BLEU or sBLEU, which has two major differences from BLEU. One is that precision p_n has no summation over hypotheses, since it is calculated for each hypothesis separately.

The other difference is that p_n for sBLEU is typically smoothed, commonly by setting initial n-gram counts to 1 (Lin and Och, 2004). This is necessary because there can often be no higher-order n-gram matches for an individual hypothesis, in which case BLEU without smoothing would become 0.0. Smoothing allows discrimination between (e.g.) a hypothesis with no matches at all and a hypothesis merely lacking 4-gram matches. However, optimizing for smoothed sBLEU can lead to short translations (Nakov et al., 2012), and does not directly match the evaluation objective of corpus-level BLEU. In Sec. 5.3 we therefore develop MRT optimization over minibatch 'corpus'-level BLEU for NMT.

2.5 Conclusions

This chapter introduces the background to Neural Machine Translation (NMT). In extremely general terms, NMT can be summarized as follows: source and target language sentences are treated as sequences of words or subword units. An end-to-end neural network architecture is defined to express a mapping between the sequences, and its parameters are adjusted according to an objective function over training data. Finally, the model translates source sentences with no provided target sentence in an inference procedure.

The vast majority of baselines in this thesis will follow the end-to-end approach described in this chapter, representing sentences with BPE subword decomposition (Sec. 2.1.2) and using a Transformer architecture (Sec. 2.2.4) trained with a cross-entropy loss function (Sec.

[2]BLEU was designed for scoring against multiple human references, but often in practice only a single reference is available. This is the case for experiments carried out in this thesis.

2.3.1), with inference conducted via beam search (Sec. 2.4.2). More detail on the specifics of baseline models are provided in the relevant experimental sections.

The approaches to NMT discussed in this chapter are not typically associated with domain adaptation. However, as part of the original contributions of this thesis we will describe the applicability of some approaches to domain adaptation, developing them further in this context:

- In Chapter 5 we further explore the effects of domain mismatch and exposure bias (Sec. 2.3) when adapting to small datasets, and develop a robust form of MRT (Sec. 2.3.1) to reduce these effects.

- In Chapter 7, we show that use of different data representations for NMT (Sec. 2.1) can benefit from many of the approaches used when translating multiple domains as defined by provenance or topic. We explore the benefits of sub-character data representations for logographic languages (Sec. 2.1.2) on higher and lower resource domains during training and inference.

- Also in Chapter 7, we develop a scheme to incorporate multiple complementary target sentence representations, particularly syntactic representations (Sec 2.1.3), in the same NMT ensemble.

The next chapter will review domain adaptation techniques for NMT. These are variations on the approaches to NMT data, architecture, training and inference presented in this chapter, intended to allow effective translation over one or more specific text domains.

Chapter 3

Domain adaptation for machine translation: a review

NMT has seen impressive advances for some translation tasks in recent years. In particular, recent WMT news and biomedical translation tasks identify several systems as performing on par with a human translator for some high-resource language pairs according to human judgements (Barrault et al., 2019; Bawden, Bretonnel Cohen, et al., 2019). Indeed, these tasks involve not only high-resource language pairs but also relatively high-resource domains, with millions of relevant sentence pairs available for training. However, NMT models perform less well on out-of-domain data. A model trained on exclusively news data is unlikely to achieve good performance on the biomedical domain, let alone human parity. Koehn and Knowles (2017) identify this 'domain mismatch' as a major challenge for NMT.

Models trained on data from *all* domains of interest can perform well across these domains. However, there is always the possibility of additional domains becoming interesting at a later stage. While it certainly possible to train a new model across all datasets from scratch for every new domain of interest, this is not generally practical. A more efficient approach is domain adaptation.

In this chapter we review prior work on domain adaptation for NMT. We first describe different elements that constitute a text domain, and then consider adaptation schemes according to the following broad areas: adaptation data selection, changes to model architecture, parameter adaptation procedure, and adaptation via inference procedure. These roughly correspond to areas of NMT model development reviewed in Sec. 2.1, 2.2, 2.3 and 2.4 respectively. We conclude the chapter by reviewing techniques for reducing gender bias in NMT as a case study for applying domain adaptation.

Domain adaptation can be targeted towards both scenarios discussed in Sec. 1.1: adaptation to a known test domain or adaptation with the possibility of an unknown test domain. In

the first case adaptation aims for optimal performance on some fixed domain without considering other domains, while in the second adaptation must achieve good performance across potentially many domains. The primary focus of this thesis is the second case, although many of the techniques reviewed are applicable in both scenarios.

3.1 What is meant by a domain?

Adapting to a 'domain' for machine translation has come to refer to a number of disparate concepts. A thorough review given in van der Wees (2017) distinguishes various aspects that combine to form a domain and studies the extent to which MT output is affected by each. In this section, we briefly review their findings and the surrounding literature, and clarify what is meant by a domain for the purposes of this thesis.

Many possible categories and sub-categories may describe a language domain for the purposes of education, text classification or data retrieval. These may not be well defined, especially across fields of research (Sinclair and Ball, 1996). However, we concentrate on the following elements of a domain as identified by van der Wees (2017) in the context of MT research:

- **Provenance:** The source of the text, usually a single discrete label. This may be a narrow description, such as the 11K English-German sentence pairs in the WMT post-editing shared task IT corpus (Turchi et al., 2017), or an extremely broad description, such as the 37M English-German web-crawled sentence pairs in the cleaned Paracrawl corpus (Bañón et al., 2020). Importantly, the provenance of a test sentence is generally unknown.

- **Topic:** The subject of text, for example news, software, biomedical. Topic often reveals itself in terms of distribution over vocabulary items. Each word in the vocabulary may have different topic-conditional probabilities, and a document (or sentence) may be classified as a mixture of topics (Blei et al., 2003). The topic(s) of a test sentence can also be determined as a mix of latent topics determined over training data. Adaptation towards such latent topics defined in terms of text features has previously been explored for statistical MT (Hasler, 2015).

- **Genre:** Genre may be interpreted as a concept that is orthogonal to topic, consisting of function, syntax and style (Santini, 2004). For example, multiple documents about a company may share topics and use similar vocabulary, such as the name of the company or specific products. However, a recruitment document, product specification, or product advertisement would all constitute different genres (Lee and Myaeng,

2002). Kessler et al. (1997) identify various cues for automatic genre detection in text, including relative frequency of different syntactic categories, use of particular characters such as punctuation marks, and sentence length distribution.

Provenance is generally unknown at test time, and topic and genre can vary between sentences or within sentences in the same document or test set. In this thesis, we refer to the domain of a machine translation model primarily in terms of either training data provenance, which is known, or measurable performance. If a model translates certain text well, we can say that the model covers the domain of the text, whether because of provenance, topic or genre coverage in the training data.

In most cases we also report test results in terms of domains by provenance, having trained and evaluated on externally defined corpora. This is necessary for evaluation campaigns, and allows us to compare easily with prior work in general. However, we develop systems by considering other elements of domain definition. In Sec. 4.3 we explore adaptation to small datasets determined by genre match as well as provenance. In Sec. 5.2 and Sec. 6.3 we adapt across a range of domains in terms of provenance, topic and genre, and show improved translation performance across test sentence provenance.

Finally, our case studies in Chapters 7 and 8 highlight low-level elements of machine translation: the effects of changing data representation on translation quality, and the effects of gender bias on coreference resolution. These have received attention as NMT research topics in their own right. However, while they can be viewed as elements of genre according to the criteria given by Kessler et al. (1997), they are not typically treated as relevant to domain adaptation. We demonstrate that combining these low-level elements of translation with domain adaptation techniques results in improved NMT performance.

3.2 Data selection for adaptation

A domain is identifiable by features of its data. Topic and genre, as described above, are often defined in terms of vocabulary choices and syntactic style. Data selection is therefore a crucial aspect of domain adaptation.

In this section we discuss selection of natural and synthetic data for domain adaptation. Most data used in training NMT models is natural – produced by a human – in which case it must usually be selected from a larger corpus according to some criteria. A special case of natural data selection is filtering for cleaner data, which is often performed before any model training. Alternatively, synthetic data can be generated for a domain, or existing in-domain data can be made partially synthetic by schemes such as noising, simplification or back translation.

3.2.1 Selecting natural data for adaptation

Given a test domain of interest and a large pool of general-domain sentences, there are many ways to extract relevant data. Sentences can be selected by word content corresponding to a topic of interest using methods developed for information retrieval, such as TF-IDF and n-gram frequency measures (Eck et al., 2004). Retrieved sentences can then be used for translation system adaptation. Similar techniques can select sentences that are *different* from existing training sentences to minimize the size of a dataset covering many domains (Eck et al., 2005), for example using Feature Decay Algorithms (Poncelas, Maillette de Buy Wenniger, et al., 2019). A related neural-specific approach is suggested by Wang, Finch, et al. (2017), who select sentences with embeddings similar to in-domain sentence embeddings to add to the in-domain corpus. Straightforward n-gram matching (Li, Zhang, et al., 2018; Zhang, Utiyama, et al., 2018) and token overlap (Xu et al., 2019) have been used to select sentence pairs for NMT adaptation to a domain defined by very few test source sentences.

Sentences can be selected after explicit scoring by external models. Moore and Lewis (2010) select data for in-doman language model training by scoring the data under in-domain and general domain language models, and taking the cross-entropy difference. This effectively scores a training sentence pair by its relevance to the in-domain corpus. Axelrod et al. (2011) add a bilingual cross-entropy difference term to select data for SMT domain adaptation specifically. However, such static cross-entropy difference filtering schemes have difficulties when in-domain and general domain corpora are very similar (van der Wees et al., 2017).

Axelrod (2017) instead suggests 'cynical data selection', an approach which repeatedly selects the sentence that most reduces the relative entropy for modelling the domain of interest. Importantly, this does not involve actually defining a domain. Indeed, Santamaría and Axelrod (2017) note that using classifiers for data selection is not necessarily a good conceptual approach since a sentence pair may easily appear in multiple corpora, and instead reframe the 'in-domain' and 'general domain' corpora as data that we know we are interested in or do not yet have an opinion about. Similarly Aharoni and Goldberg (2020) dispense with assigned corpus labels and show that adapting to data identified by unsupervised domain clustering using large language models matches or out-performs tuning on the 'correct' domain-labelled data. In Sec. 5.2 we explore the concept of overlapping domain data for NMT model adaptation between small domains as defined by provenance.

Data filtering

By far the simplest approach to natural data selection for adaptation is use of provenance. This means, where possible, taking an existing relevant corpus label as indicative of domain. In this case data filtering can still be applied to ensure that all selected data is actually representative of a domain. For example, Taghipour et al. (2011) map sentences in a constructed parallel corpus to a feature space, and mark the most novel pairs in the feature space as noise to be removed. Some care must be taken not to diminish the training space too far: for example, Lewis and Eetemadi (2013) attempt to maximize n-gram coverage with remaining data while filtering sentences for SMT.

A special case of data filtering is targeted to remove 'noisy' training examples. This may be applied to a general domain corpus before a model is trained to ensure that source and target sentences in the training sentence are well-aligned, contain the languages of interest, are not too long or contain too many non-words (e.g. HTML tags) (Berard et al., 2019; Khayrallah and Koehn, 2018). This can be achieved in very similar ways to domain data selection, for example with extensions of cross-entropy difference filtering (Moore and Lewis, 2010) to bilingual training examples, where the 'in-domain' models are trained on clean data only (Junczys-Dowmunt, 2018a; Junczys-Dowmunt, 2018b).

3.2.2 Generating synthetic data for adaptation

Bilingual training data that is relevant to the domain of interest may not be available. In this case, it is often possible to construct partially or fully synthetic bilingual training corpora. This is a case of data *generation* for adaptation rather than data selection. Partially synthetic data may be monolingual data that is forward- or back-translated to form bitext, or existing bilingual data that is noised or simplified. Completely synthetic data may be template-generated, or involve an external or induced lexicon.

Back translation and forward translation

Given monolingual data in the domain of interest and a sufficiently strong existing NMT model, it is possible to back-translate or forward-translate the monolingual data to obtain aligned source and target language training sentences.

Back translation treats the natural monolingual data as target sentences, and requires a target-to-source NMT model to generate synthetic source sentences. Back translation can be applied to a specific domain but is very commonly used to augment general domain translation corpora, with strong improvements over models not trained on back-translated data (Sennrich, Haddow, and Birch, 2016c). Variations on back translation include sampling

multiple source sentences instead of back-translating with beam search (Imamura et al., 2018) or noising back translations (Edunov et al., 2018b). Models trained primarily or even exclusively on back translated data have shown similar performance to models trained on natural data(Poncelas, Shterionov, et al., 2018). Back translation has been shown to out-perform forward translation in the context of domain adaptation for SMT (Lambert et al., 2011), while using back translated data for NMT adaptation can require synthetic data 'repair' to reduce noise (Wei et al., 2020).

Forward translation treats the natural data as source sentences and generates synthetic target sentences with an existing source-to-target NMT model. While forward translation is less widely used than back translation, it can be used for efficient domain adaptation. For example, a single model can forward-translate to generate synthetic bitext in the domain of interest and then can itself be adapted to that bitext, as in Chinea-Ríos et al. (2017), while back translation requires an additional reverse NMT model. A variation uses a much larger or otherwise stronger 'teacher' model to generate in-domain forward translations which are then used to train or tune a 'student' model (Currey, Mathur, et al., 2020; Gordon and Duh, 2020).

Unlike back translation, forward translation can also be applied to source data with particular characteristics, such as the test set itself. This results in either synthetic test target sentences, or synthetic-target test bitext. Synthetic target sentences alone may be used as a seed to retrieve more relevant natural or synthetic bitext for adaptation (Poncelas, Wenniger, et al., 2018). Synthetic-target bitext can be used for further training directly, or introduced into a corpus of candidate sentences from which it may be selected using a domain-specific scheme (Poncelas and Way, 2019). In Sec. 8.3 we investigate forward translation in the context of counterfactual data augmentation for direct domain adaptation.

Artificially-noising and simplifying natural data

An alternative way to generate additional synthetic data is to take existing natural bitext and change the source or target sentence in some way. A common example is adding artificial noise to source language training sentences. Source language characters and words can be deleted, substituted or permuted. Training on these adversarial synthetic examples have been shown to improve robustness to natural noise in test sentences (Karpukhin et al., 2019; Vaibhav et al., 2019). Additionally Tan et al. (2020) demonstrate improved NMT robustness to linguistic variation by fine-tuning on synthetic adversarial examples.

Synthetic examples can also be constructed from natural text by simplifying some source (Hasler, de Gispert, Stahlberg, et al., 2017; Li, Wang, et al., 2020) or target (Agrawal and Carpuat, 2019) sentences before training. The former case can make sentences easier to

translate and can be applied to test sentences. The latter allows translation into language with a specified complexity level. However, sentence simplification approaches are less common than noising approaches due to the difficulty in obtaining simplification training data and models, which may themselves rely on the presence of large-scale corpora for a given language or be domain-sensitive (Mehta et al., 2020). These are not needed if the goal is noising: it is intuitively more difficult to correctly simplify natural language than to add errors to it.

One motivation for including synthetic examples is improving robustness to noisy training examples. We note that any synthetic variations on existing bilingual examples may also cause a regularization effect by reducing the likelihood of over-fitting to a small set of one-to-one training examples (Bishop, 1995).

Purely synthetic data for adaptation

A final genre of data for adaptation is purely synthetic data. This may be obtained from an external or induced lexicon, or constructed from a template.

Lexicons have been used effectively when dealing with rare words, OOV words or ambiguous words with multiple senses in the training data (Zhao, Zhang, et al., 2018). For SMT lexicons can be used to mine translation model probabilities directly (Daumé III and Jagarlamudi, 2011). In NMT lexicon probabilities may be incorporated into the NMT loss function (Arthur et al., 2016) or bilingual lexicon entries may be used to construct partially synthetic sentence pairs (Zhang and Zong, 2016).

Another application of synthetic lexicon data is words or phrases for which there is an easily obtainable translation, and that the model is likely to be required to translate. This type of data is usually domain specific: NMT for use on social media may require translations for common greetings, while a biomedical NMT system may have a fixed set of terminology to translate. Hu et al. (2019) adapt to a lexicon for a given domain, while Kothur et al. (2018) adapt to a lexicon containing novel words in a test document. Song, Zhang, et al. (2019) replace certain source phrases with pre-specified target translations to encourage copy behaviour. In Sec. 8.3 we discuss creation of synthetic sets of adversarial examples in the context of mitigating the effects of gender bias in NMT.

3.3 Architecture-centric adaptation approaches

Architecture-centric approaches to domain adaptation typically add trainable parameters to the NMT model itself. This may be a single new layer or domain discriminator, or a new subnetwork. These parameters may be determined when the model is first defined, or added

before tuning. The aim is generally to improve model performance over some identifiable new domain.

One genre of architecture-based approach treats the encoding or decoding procedure differently depending on domain. Zeng et al. (2018) and Pham et al. (2019) change word embeddings to have domain-specific features, while Gu, Feng, et al. (2019) use a combination of shared and domain-specific encoders and decoders. Nguyen and Chiang (2018) augment models with a lexical choice network targeted at improving translation of rare or ambiguous words in a given domain. Wang, Wang, et al. (2020) augment NMT models with networks to identify domain-specific features.

Where data is domain-labelled, the labels themselves can be used to signal domain for a multi-domain system (Kobus et al., 2017). This approach can be scaled to new domains by adding more labels (Tars and Fishel, 2018). Architectural changes can then take advantage of these labels. Britz, Le, et al. (2017) share encoders and decoders across domains but add a domain discriminator to determine a target domain label corresponding to one of the training domains.

A lightweight approach to domain adaptation for NMT adds only a limited number of parameters. The added parameters are adapted only on in-domain data, and pre-trained parameters may be 'frozen' – held at their pre-trained values. This is the approach taken by Vilar (2018), who effectively regularizes parameters with an added importance network, and Bapna and Firat (2019), who freeze general-domain models and add small adapter layers for fine-tuning on a given domain.

Such architectural approaches are capable of good performance over multiple domains. If original parameters are left unchanged and only a new set of parameters is adapted, performance degradation on the original domain can avoided by simply using the original parameters. However, such approaches implicitly assume that language domains are discrete, distinct entities. New architecture may be either 'activated' if the test set is in-domain, or 'deactivated' for better general domain performance (Vilar, 2018). A sentence may be assigned to a single domain, and a label added for that domain. (Tars and Fishel, 2018).

By contrast, this thesis takes the view that multiple text domains may overlap, and that training domains may be mutually beneficial for translation performance. This is particularly likely given that the unknown test data domain may not be an exact mapping to one of the training domains, but may also be the case for 'known' domain sentences which benefit from the general translation model. The rest of this thesis therefore discusses schemes for adapting an NMT model without changes to the architecture.

3.4 Training schemes for adaptation

Once data is selected or generated for adaptation, a pre-trained model can be fine-tuned on that data. Fine-tuning on data involves model parameter adaptation according to loss on that data. Straightforward continuation of training on the new data is the simplest approach. However, this often leads to over-fitting on the new data and catastrophic forgetting of performance on previous domains (McCloskey and Cohen, 1989; Ratcliff, 1990). This section will review the catastrophic forgetting effect in the context of NMT domain adaptation, as well as training schemes proposed to mitigate it: regularization, curriculum learning and instance weighting.

3.4.1 Fine-tuning and catastrophic forgetting

An extremely simple way to adapt to a new domain is to continue training a pre-trained general domain model on a smaller amount of data from the domain of interest (Luong and Manning, 2015). A special case of fine-tuning for adaptation occurs when the new data has a different source language (Zoph et al., 2016) or target language (Kocmi and Bojar, 2018) from the original model. This is common when developing NMT systems to translate low-resource language pairs. A distinction is that in this case catastrophic forgetting of the original model's abilities is less of a concern, since it is likely that a user can pre-determine which languages will be translated with which model. However, many techniques are applicable to both single-language-pair domain adaptation and cross-lingual transfer learning.

For domain adaptation, catastrophic forgetting is less important when the model is intended to translate only a small amount of highly specific data. Examples include adapting a new model to translate each individual test sentence (Li, Zhang, et al., 2018) or document (Kothur et al., 2018). In Sec. 5.2 we discuss fine-tuning for a biomedical scenario where the target domain is known and quite narrow. However, in general, the pre-trained model has some translation ability which it is preferable not to forget.

A straightforward approach to avoiding catastrophic forgetting is to simply fine-tune for fewer steps (Xu et al., 2019). However, this introduces an inherent trade-off between better performance on the new domain and worse performance on the old domain. Other approaches to good multi-domain performance with a single model include parameter regularization (Sec. 3.4.2), changing the order that data is shown to the model (curriculum learning, Sec. 3.4.3) or the impact data has on a model (instance weighting, Sec. 3.4.4).

3.4.2 Parameter regularization

A straightforward way to avoid forgetting is to minimize changes to the model parameters. The intuition is that if parameters stay close to their pre-trained values, they will give similar performance on the pre-training domain. For example, Thompson, Khayrallah, et al. (2018) and Michel and Neubig (2018) simply choose subsets of the NMT model parameters to hold at their pre-trained values when fine-tuning on a new domain. Wuebker et al. (2018) likewise adapt only a subset of model parameters, encouraging sparsity in the adapted parameters with L1 regularization to improve efficiency.

Barone et al. (2017) allow all NMT model parameters to vary under L2 regularization relative to their pre-trained values θ^{PT}. Kirkpatrick et al. (2017) introduce the related approach of Elastic Weight Consolidation (EWC) for computer vision domain adaptation. EWC effectively scales the L2 regularization applied to each parameter θ_j by F_j. We illustrate these approaches in Figure 3.1. If Λ is a scalar weight a general-form L2 regularized loss function is:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}}[L_{CE}(\mathbf{x},\mathbf{y};\theta) + \Lambda \sum_j F_j(\theta_j - \theta_j^{PT})^2] \tag{3.1}$$

Relative domain importance can be controlled or tuned with Λ, which is larger if the old domain is more important and smaller if the new domain is more important. L2 regularization occurs where $F_j = 1$ for all j. For EWC Kirkpatrick et al. (2017) define F_j as the Fisher information for the pre-training domain estimated over a sample of data from that domain, $(\mathbf{x}^{PT}, \mathbf{y}^{PT})$.

$$F_j = \mathbb{E}\left[\nabla^2 L_{CE}(\mathbf{x}^{PT}, \mathbf{y}^{PT}; \theta_j^{PT})\right] \tag{3.2}$$

In Sec. 5.2 we describe our own experiments with EWC and L2 for NMT domain adaptation (Saunders, Stahlberg, de Gispert, et al., 2019). Independently and concurrently with our work, Thompson, Gwinnup, et al. (2019) also apply EWC to reduce forgetting during NMT domain adaptation.

Knowledge distillation

Knowledge distillation and similar 'teacher-student' model compression schemes effectively use one teacher model to regularize training or tuning of a separate student model (Buciluǎ et al., 2006; Hinton, Vinyals, et al., 2015). Typically the teacher model is a large, pre-trained model, and the student is required to emulate its behaviour with far fewer parameters. The student model is fine-tuned such that its *output distribution* remains similar to the teacher model's output distribution over the pre-trained data (Kim and Rush, 2016). A distinct but

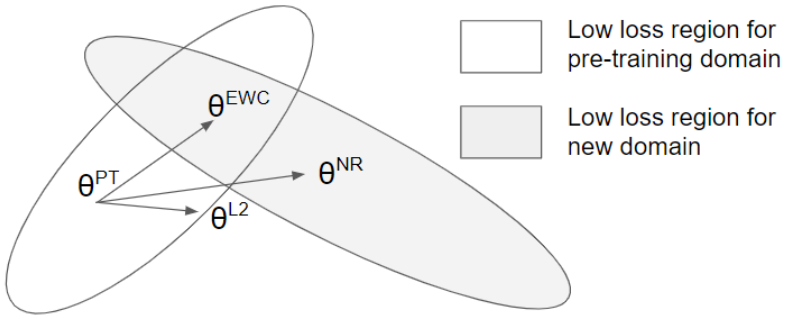


Fig. 3.1 Illustration of parameter regularization during fine-tuning from a pre-training (PT) domain to a new domain, based on Fig. 1 from Kirkpatrick et al. (2017). If the pre-trained parameters θ^{PT} are adapted with no regularization (NR), good performance on the new domain corresponds to catastrophic forgetting on the pre-training domain. Applying the same regularization to all parameters (L2) encourages minimal overall change from θ^{PT}, which may not allow sufficient improvement on the new domain. EWC regularization (Kirkpatrick et al., 2017) aims to allow good performance for the new domain by varying the parameters that are unimportant for the pre-training domain.

related approach simply prunes the large teacher model's parameters while attempting to maintain performance (LeCun et al., 1990; Voita, Talbot, et al., 2019).

In a domain adaptation context, knowledge distillation encourages similar performance on the pre-training domain with a regularization function between general and in-domain model output distributions (Dakwale and Monz, 2017; Khayrallah, Thompson, et al., 2018; Mghabbar and Ratnamogan, 2020). We see this approach as similar in spirit to parameter regularization while being more complex, since two models must actually operate on the data. This can be effective when the aim is to compress the teacher model, since in this case the in-domain student model is likely to be much smaller than the other. For models remaining the same size we view parameter regularization as more practical.

3.4.3 Curriculum learning

Bengio, Louradour, et al. (2009) recognize that humans learn best when concepts are presented in a meaningful order, or a curriculum. They hypothesize that neural model training can benefit from the same strategy of curriculum learning in terms of convergence speed or quality of converged model, and demonstrate that this is the case for a language modelling task with a curriculum of increasing vocabulary size.

In broad terms, a curriculum ranks the training examples. The ranking then guides the order in which examples are presented to the model during training or fine-tuning. A typical ranking when applying a curriculum throughout training orders training examples by

difficulty, with the easiest examples shown first and the more complex examples introduced later (Weinshall et al., 2018; Zhang, Kim, et al., 2017). Difficulty can be determined in terms of data features like sentence length, linguistic complexity or word rarity (Kocmi and Bojar, 2017; Platanios et al., 2019).

Difficulty can also be based on the 'competence' of the model (Platanios et al., 2019). A training example may be considered difficult for an NMT model at a given point in training if the example's embedding norm is large (Liu, Lai, et al., 2020), if the training loss of the example is changing significantly between training iterations (Wang, Utiyama, and Sumita, 2018), or if the model simply does not translate it well (Dou et al., 2020).

Much of the above work on curriculum learning frames the problem as discovering an 'easiest-to-hardest' ranking. Other rankings are possible. In fact, Zhang, Kumar, et al. (2018) find that both easy-first and hardest-first schedules can give similar convergence improvements. Another form of curriculum learning for NMT which does not depend on sample difficulty is scheduled sampling (Bengio, Vinyals, et al., 2015). Scheduled sampling gradually anneals target sequences during training from human references to sequences partially generated by the NMT model in an effort to avoid exposure bias (Sec. 2.3).

More relevant to this thesis, a curriculum ranking can be constructed from least-domain-relevant examples to most-domain-relevant. In fact, simple fine-tuning (Sec 3.4.1) can be seen as an example of curriculum learning where the final part of the curriculum contains only in-domain data. Curriculum schemes may therefore have much in common with data-centric adaptation methods. However, curriculum-based approaches to domain adaptation generally involve a gradual transition to in-domain data. For example, Wang, Watanabe, et al. (2018) use an incremental denoising curriculum to fine-tune a pre-trained NMT model on increasingly clean data from its existing training corpus. Similar 'cleaning' fine-tuning curricula data can be learned via reinforcement learning methods (Kumar, Foster, et al., 2019; Zhao, Wu, et al., 2020).

When adapting to a distinct domain a curriculum can be determined in terms of similarity score with known target domain data. The scored data may be 'pseudo in-domain' data extracted from general corpora that has been previously seen by the model (Farajian et al., 2017; van der Wees et al., 2017; Zhang, Shapiro, et al., 2019), some mixture of in-domain and general-domain data (Chu et al., 2017; Sajjad et al., 2017), or data from multiple domains identified with domain-specific features (Wang, Tian, et al., 2020). Incorporating previously-seen data into the curriculum can be used to reduce forgetting: for example, Aljundi et al. (2019) demonstrate that maintaining a representative 'replay buffer' of past training examples can avoid forgetting even when hard domain boundaries are not available.

3.4.4 Instance weighting

Instance weighting is a scheme where training examples are weighted according to their relevance to the target domain (Foster et al., 2010). For NMT, an instance weight $W_{x,y}$ for each source-target training example can easily be integrated into a cross-entropy-based loss function:

$$L(\mathbf{x},\mathbf{y};\theta) = \sum_{(\mathbf{x},\mathbf{y})} -W_{\mathbf{x},\mathbf{y}} \log P(\mathbf{y}|\mathbf{x};\theta) \tag{3.3}$$

A higher weight indicates that a sentence pair is more important for training towards the target domain, while a low (or zero) weight will lead to sentences effectively being ignored for training purposes. The weight may be determined in various ways. It may be the same for all sentences marked as from a given domain, or defined for each sentence using a domain similarity measure like n-gram similarity (Joty et al., 2015) or cross-entropy difference (Wang, Utiyama, Liu, et al., 2017). If changes can be made to the model architecture, the instance weight may be determined by a domain classifier (Chen, Cherry, et al., 2017), or an architecture-dependent approach like sentence embedding similarity (Zhang and Xiong, 2018).

We view instance weighting as fundamentally the same idea as curriculum learning (Sec 3.4.3). Both schemes bias the model to place more importance on losses for certain training examples. Some forms of curriculum learning are implemented in a similar way to instance weighting, with a higher weight applied to examples that fall into the current section of the curriculum, or a zero weight applied to examples that should not yet be shown to the model (Bengio, Louradour, et al., 2009; Dou et al., 2020). One difference is that instance weights for domain adaptation do not usually change as training progresses or model competence changes, but bias the model towards in-domain data in a constant manner.

3.5 Inference schemes for adaptation

One way to side-step the problem of catastrophic forgetting is simply assigning a separate NMT model to each domain. Such models can be obtained using the techniques discussed in all previous sections of this chapter, for example by fine-tuning a single pre-trained model on data from each domain of interest.

While this approach is simple, if not memory-efficient, it begs the question of how best to perform translation on an unseen source sentence from an unknown domain. Two possible approaches are multi-domain ensembling, and reranking or rescoring an existing set of translation hypotheses.

3.5.1 Multi-domain ensembling

At inference time an NMT ensemble can use the predictions of multiple models to produce a translation in a single pass, as described in Sec. 2.4.3. It may be that in a given scenario certain models in an ensemble are more useful than others. For example, integration of a language model could provide improved fluency (Gulcehre et al., 2017; Stahlberg, Cross, et al., 2018). NMT models which use different surface-level representations of the source sentence (Hokamp, 2017) or translate the source sentence from different languages (Firat et al., 2016; Garmash and Monz, 2016) may likewise contribute differently to a translation depending on the source sentence. In Sec. 7.3 we explore schemes for ensembling multiple NMT models with different surface-level representations of the *target* sentence, and describe the potential benefits of doing so (Saunders, Stahlberg, de Gispert, et al., 2018).

In the context of a source sentence of unknown domain, Freitag and Al-Onaizan (2016) show good performance using ensembles of general models and in-domain models fine-tuned without regularization. Sajjad et al. (2017) also use multi-domain ensembles and weight the contribution of each ensemble model as in Eq. 2.18. They determine static ensemble weights tuned on development sets. For SMT, (Huck, Birch, et al., 2015) use a language model to classify the domain of a test sentence when determining which set of parameters to use when generating the translation hypothesis.

Allauzen and Riley (2011) introduce Bayesian Interpolation (BI) for adaptively weighting language models in ensembles for speech recognition. Importantly, they do not necessarily specify that the 'task', t – the domain of the test sentence – corresponds to exactly one of the model domains k. Neither do they assume that a uniform weighting of all K domains is optimal for all tasks. Instead they use a set of tuned ensemble weights $\lambda_{k,t}$, which defines a task-conditional ensemble:

$$p(\mathbf{y}|t) = \sum_{k=1}^{K} \lambda_{k,t}\ p_k(\mathbf{y}) \tag{3.4}$$

This can be used as a fixed weight ensemble if task t is known. However if t is not known, the ensemble can be written as follows at inference step i, where h_i is history $\mathbf{y}_{1:i-1}$:

$$\begin{aligned} p(y_i|h_i) &= \sum_{t=1}^{T} p(t, y_i|h_i) \\ &= \sum_{t=1}^{T} p(t|h_i)\, p(y_i|h_i, t) \\ &= \sum_{k=1}^{K} p_k(y_i|h_i) \sum_{t=1}^{T} p(t|h_i)\lambda_{k,t} \\ &= \sum_{k=1}^{K} W_{k,i}\, p_k(y_i|h_i) \end{aligned} \tag{3.5}$$

That is, a weighted ensemble with state-dependent mixture weights computable from task priors and the updated language model task posterior:

$$p(t|h_i) = \frac{p(h_i|t)p(t)}{\sum_{t'=1}^{T} p(h_i|t')p(t')} \tag{3.6}$$

In Sec. 6.3 we extend this formalism to include conditioning on a source sentence. This lets us apply Bayesian Interpolation to domain adaptive NMT with multi-domain ensembles for situations where the test sentence domain is unknown (Saunders, Stahlberg, de Gispert, et al., 2019).

3.5.2 Constrained inference and rescoring

Ensembling uses multiple models to produce a translation simultaneously. Another option is to produce an initial translation with a single model, then adjust or 'correct' the translation using another model. This is at minimum a two-step process, since multiple models must perform an inference pass. However, it can be more efficient than ensembling: rescoring does not involve holding multiple models in memory at once, and the second translation pass is commonly held close to the initial translation in some way.

If the initial model produces multiple translations – for example, the highest scoring N translations following beam search (Sec. 2.4.2) – a typical approach to multi-pass inference is rescoring this N-best list using a different model or loss function. For example, MBR decoding rescores N-best lists or lattices to improve single system performance for SMT (de Gispert et al., 2010; Kumar and Byrne, 2004; Tromble et al., 2008), or for NMT if a sufficiently diverse lattice can be defined, for example, from SMT n-gram posteriors (Stahlberg, de Gispert, Hasler, et al., 2017). A neural-only approach to this problem rescores

the N-best list of a L2R NMT model with a R2L NMT model (Liu, Utiyama, et al., 2016; Sennrich, Haddow, and Birch, 2016b).

A related idea is constrained inference. For example, Stahlberg, Hasler, Waite, et al. (2016) generate translation hypotheses with an SMT model, which are then represented as a lattice that constrains NMT decoding. Khayrallah, Kumar, et al. (2017) constrain NMT output to an SMT lattice to improve adequacy in domain adaptation scenarios. The initial translation may itself be constrained: for example Hasler, de Gispert, Iglesias, et al. (2018) constrain the NMT output to ensure it produces given terminology. Outside of NMT, lattice-constrained rescoring of neural models has been applied to grammatical error correction (GEC) (Stahlberg, Bryant, et al., 2019) and speech recognition (Liu, Chen, et al., 2016). In Sec. 8.4.1, we describe the construction of gender-inflected search spaces for rescoring to mitigate bias effects in translation.

We note that constrained lattice search does not require the rescoring system to use the same target representation as the original system. For example, Ragni et al. (2017) perform speech recognition and keyword search on morph-based initial lattices with word-based models using an intermediary morph-to-word transduction. In Sec. 7.3 we use lattice-constrained rescoring with intermediary transduction lattices to allow NMT model ensembles with multiple target representations.

3.6 Gender bias in machine translation as a case study for multi-domain adaptation

One of the novel contributions of this thesis is the framing of gender bias in NMT as a domain adaptation problem. We therefore conclude this literature review with an overview of the gender bias problem in NMT and prior work pertaining to it, with some motivation for considering it a multi-domain adaptation problem.

3.6.1 Problem background

Translation into languages with grammatical gender involves correctly inferring the grammatical gender of all entities in a sentence. In some languages this grammatical gender is dependent on the social gender of human referents. For example, in German, translation of the entity 'the doctor' would be feminine for a female doctor – *Die Ärztin* – or masculine for a male doctor – *Der Arzt*.

[1]Machine translation from Google Translate 7 Sept. 2020

English source	The **doctor** helps the patient, although **she** is busy
German reference	**Die Ärztin** hilft dem Patienten, obwohl **sie** beschäftigt ist *The [female] doctor helps the [male] patient, although she is busy*
MT with a bias-related mistake[1]	Der Arzt hilft **der Patientin**, obwohl **sie** beschäftigt ist *The [male] doctor helps the [female] patient, although she is busy*
English source	The **nurse** helps the patient, although **he** is busy
German reference	**Der Krankenpfleger** hilft dem Patienten, obwohl **er** beschäftigt ist *The [male] nurse helps the [male] patient, although he is busy*
MT with a bias-related mistake	Die Krankenschwester hilft **dem Patienten**, obwohl **er** beschäftigt ist *The [female] nurse helps the [male] patient, although he is busy*

Table 3.1 Two examples of mistranslation relating to gender bias effects. Bolded words are entities inflected to correspond to the pronoun. In the English source sentences 'the doctor' is coreferent with the pronoun 'she' and should be feminine-inflected, and 'the nurse' is coreferent with 'he' and should be masculine-inflected. In the first example the machine translation wrongly inflects the doctor entity as masculine and the patient entity as feminine. In the second example the nurse is wrongly feminine-inflected even though the sentence has no feminine pronoun.

In practice, however, many NMT models struggle at generating such inflections correctly (Prates et al., 2019). Gender-based errors are particularly common when translating coreference resolution sentences with two entities, only one of which is coreferent with a pronoun (Stanovsky et al., 2019). Table 3.1 gives two typical examples. In the first the machine translation system incorrectly inflects the German hypothesis to contain a masculine doctor and feminine patient, even though the 'doctor' is the entity coreferent with the feminine pronoun. In the second it incorrectly inflects the German hypothesis to contain a feminine nurse, even though the 'nurse' is the entity coreferent with the masculine pronoun, and there is no feminine pronoun in the source sentence.

Stanovsky et al. (2019) explore these mistakes and demonstrate that they tend to reflect social gender bias: machine translation tends to translate based on profession-based gender stereotypes instead of correctly performing coreference resolution and translating using this meaningful context. This may be because the systems are influenced by the higher frequency of masculine-inflected doctors and feminine-inflected nurses in training data. We term this a gender bias effect.

3.6.2 Reducing the effects of gender bias in NMT

In recent years there has been much interest in reducing the effects of gender bias in NMT output. These fall broadly into two categories (Sun, Gaut, et al., 2019). The first consists of work that attempts to control or 'balance' the training data or the model's word embeddings

to reduce the likelihood of translations exhibiting bias. The second seeks to control gender inflection in the target language by explicitly or implicitly adding gender features during training or inference which the model can rely on instead of any preconceptions.

Addressing bias with data balancing

Recent recommendations for ethics in Artificial Intelligence have suggested that social biases or imbalances in a dataset be addressed prior to model training (HLEG, 2019). This recommendation makes some intuitive sense: if bias stems from imbalances in the data, a natural response would be attempting to remove these imbalances. Common related approaches are counterfactual data augmentation and word embedding debiasing.

Counterfactual data augmentation involves identifying the subset of sentences containing bias – in this case gendered terms – and, for each one, adding an equivalent sentence with the bias reversed – in this case a differently gendered version (Lu et al., 2020). Zhao, Wang, et al. (2018) show improvement in gender coreference resolution for English by training on counterfactually augmented data. Zmigrod et al. (2019) demonstrate a more complicated scheme for gender-inflected languages.

An alternative approach is proposed by Bolukbasi et al. (2016), who identify bias in embeddings in terms of a gender subspace. Sets of gendered words such as pronouns or professions have clustered word embeddings. The direction of maximum variation in these embeddings is interpreted to constitute gender bias. Flattening this direction in the embeddings either during training or before inference can reduce some effects of bias, either binary or multi-class (Manzini et al., 2019). Escudé Font and Costa-jussà (2019) train NMT models from scratch with debiased word embeddings, demonstrating improved performance on an English-Spanish occupations task with a single profession and pronoun per sentence.

We highlight two major difficulties with data-balancing schemes for gender bias reduction in NMT. The first is in how far they apply to the real problem. The recommendation of pre-training bias removal from HLEG (2019) presupposes that the source of bias in a dataset is both obvious and easily adjusted. In fact there are countless ways in which biases could conceivably manifest in generated natural language, relating to gender or otherwise (Hovy et al., 2020; Shah et al., 2020), so speaking in terms of simple biases or imbalances that can be addressed is not clearly meaningful.

The second difficulty is practical. In terms of debiasing word embeddings, there is some evidence that these techniques have only superficial effects, since embeddings for words with similar biases are still clustered even if a 'bias' direction is flattened (Gonen and Goldberg, 2019). In terms of data balancing, attempts to gender-balance even monolingual data in inflected languages struggle with multiple-entity sentences like those in Table 3.1 (Zmigrod

et al., 2019). The difficulty is compounded for the large bilingual corpora required to train NMT models. In Sec. 8.3 we discuss the challenges involved in even a simple scheme for approximate counterfactual data augmentation for NMT.

Finally, all of these approaches involve training the model from scratch. Practically speaking this is very inefficient, particularly when new sources of bias may be identified at a later stage or introduced through the retraining process itself.

Addressing bias with context-based features

The idea of controlling machine translation gender inflections with a tag or signal has been proposed in several forms. Vanmassenhove et al. (2018) incorporate a 'speaker gender' tag into training data, allowing gender to be conveyed at the sentence level. However, this does not allow more fine-grained control, for example if there is more than one referent in a sentence. Similar approaches from Voita, Serdyukov, et al. (2018) and Basta et al. (2020) infer and use gender information from discourse context. Moryossef et al. (2019) also incorporate a single explicit gender feature for each sentence for inference. Miculicich Werlen and Popescu-Belis (2017) integrate coreference links into machine translation reranking to improve pronoun translation with cross-sentence context. Stanovsky et al. (2019) propose NMT gender bias reduction by 'mixing signals' with the addition of pro-stereotypical adjectives. Stafanovičs et al. (2020) use a fine-grained approach in training NMT models from scratch with all source language words annotated with target language grammatical gender.

Gender tagging is often effective at controlling the gender inflection of translations, and tagging can be applied before training or at inference time. However, tags are typically not introduced during fine-tuning, and existing work typically only measures the change in performance on the intended targets of gender tags. In Sec. 8.5 we incorporate new gender tags during fine-tuning and investigate some unintended consequences of gender tagging methods, as well as proposing mitigating techniques.

3.6.3 Gender bias in NMT as a multi-domain adaptation problem?

The prior work reviewed in this section has almost exclusively involved retraining a model from scratch. However, we consider the problem of reducing gender bias effects in NMT to be an excellent candidate for multi-domain adaptation techniques:

- We wish to improve translation performance on sentences with a distinct vocabulary distribution, which can be interpreted as a domain: sentences containing gendered

terms which do not correspond to existing social biases, such as female doctors and male nurses.

- We want the ability to continually adapt to new 'domains' in order to reduce the effect of newly identified bias sources, since it is impossible to pre-determine all possible sources of harmful bias.
- We wish to avoid retraining from scratch.
- We wish to keep general translation ability obtained from the training on the biased 'pre-training' domain.
- We may have to translate test sentences that do not belong to the new 'domain'. That is, a given sentence may or may not have any gendered terms to translate that are affected by gender bias effects.

In Chapter 8 we apply many of the domain adaptation techniques reviewed previously in this chapter to mitigate the effects of gender bias on NMT.

3.7 Conclusions

Domain adaptation allows NMT models to achieve good performance on language of interest with very limited training data, and without the cost of training the model from scratch. Adaptation may even allow better performance than from-scratch training on a given domain, or on sentences whose domain is unknown.

The remainder of this thesis describes original contributions on the topic of domain adaptation for NMT. We demonstrate that careful approaches, whether data-centric or involving more significant changes to the adaptation and inference procedures, can permit strong improvements on new or unknown translation domains without requiring retraining from scratch or changes to model architecture. (Sec. 3.3, which discusses approaches that do involve architecture changes, is included for completion). Many of the contributions follow on directly from approaches touched on in this chapter (as well as those following on from the more general NMT literature review in the previous chapter, described in Sec. 2.5):

- In Chapter 4 we explore data-centric schemes for domain adaptation (Sec. 3.2). We highlight the advantages and disadvantages of varying data only in domain adaptation.
- In Chapter 5 we explore fine-tuning schemes for NMT domain adaptation to mitigate the catastrophic forgetting problem during domain adaptation (Sec. 3.4). We investigate the use of L2 and EWC regularization during fine-tuning to avoid forgetting.

- In Chapter 6 we build on inference techniques for domain adaptation (Sec. 3.5), extending Bayesian Interpolation to source sentence dependence for domain adaptive ensembling.
- In Chapter 7 we extend the concept of multi-domain NMT model ensembling (Sec. 3.5) to explore the benefits of ensembles with multiple target representations.
- Chapter 8 frames the gender bias problem in NMT (Sec. 3.6) as a domain adaptation problem, and explores potential solutions using domain adaptation techniques reviewed in this chapter: synthetic and partially synthetic dataset construction (Sec. 3.2), regularized and tagged adaptation (Sec. 3.4) and constrained rescoring (Sec. 3.5).

Chapter 4

Data-centric approaches to domain adaptation

This chapter draws from the following publications: Saunders, Stahlberg, and Byrne (2019) in Sec. 4.2, and Saunders and Byrne (2020a) in Sec. 4.3.

4.1 Motivation

As discussed in Sec. 3.1 and Sec. 3.2, a domain can be described by its data. A sentence may contain elements of one or more topics or genres in terms of vocabulary choice or structural aspects. The domains of sentences available during training or fine-tuning will affect how fast the NMT model parameters converge, and the local optimum to which they converge. During inference, the translation quality for a given test sentence will depend on whether the model has trained on sentences with similar or different domains.

In this chapter we explore data-centric approaches to domain adaptation in the context of our submissions to two consecutive years of the WMT biomedical translation task. We wish to address the first research question raised in Sec. 1.1.1, exploring the effectiveness of straightforward data-centric approaches to domain adaptation, particularly with regards to domain robustness and possible side-effects. A large body of existing research explores data selection and generation for NMT adaptation, as reviewed in Sec. 3.2. Rather than reproduce such investigations, and since the biomedical task test sets have a known topic[1] and genre[2],

[1] Biomedical sciences, although this itself covers a range of other topics: medical descriptions, numerical results, historical or geographical asides to provide context, etc.

[2] Medline paper abstracts. Detailed descriptions and sources are given in Bawden, Bretonnel Cohen, et al. (2019) and Bawden, Di Nunzio, et al. (2020)

we select fine-tuning data-sets by provenance to illustrate the effects of fine-tuning on either large, topic-relevant corpora or a small, genre-matched corpus.

In Sec. 4.2 we describe experiments with the widely-used approach of transfer learning to a new corpus. While treating the provenance of a whole corpus as an accurate indication of domain for all sentences in the corpus, we show that an iterative approach to transfer learning across related domains can improve performance. However, we also demonstrate side-effects in terms of 'forgetting' of previously learned domains, even those closely related to the fine-tuning domain.

In Sec. 4.3 we explore the impact of adapting to a small subset of the in-domain training data which is particularly similar in genre to a test dataset. We find that this can still further improve translation quality in some cases, but risks over-exposing the model to degeneracies in the training data, resulting in exposure bias.

4.2 Iterative transfer learning and the WMT19 biomedical translation task

This section discusses our participation in the WMT19 biomedical translation task[3]. NMT in the biomedical domain presents challenges in addition to general domain translation, as for many small domains. Available corpora are relatively small, exacerbating the effect of noisy or poorly aligned training data. Vocabulary can be very topic-specific, so training to convergence on a single biomedical dataset may not correspond to good performance on arbitrary biomedical test data. Instead we focus on building strong models over multiple related domains using iterative transfer learning. These single-domain models can then be combined in an ensemble during inference.

In this section we treat the provenance of a corpus as indicative of its domain, and adapt NMT models simply by continuing training the model parameters on labelled corpora. We use our submissions to the WMT19 biomedical task to demonstrate that the effectiveness of this apparently straightforward scheme is heavily dependent on data domain. We find that even ostensibly uniform domains can be sub-divided into more specific domains such that performance on each sub-domain is measurably different. We attempt to leverage this domain decomposability to improve domain adaptation purely by selecting and ordering fine-tuning corpora from different domains, effectively defining a domain curriculum.

[3]http://www.statmt.org/wmt19/biomedical-translation-task.html

4.2.1 Iterative transfer learning

Transfer learning for domain adaptation typically involves initial training on a large, general domain corpus, followed by fine-tuning on the domain of interest. Transfer learning of this kind is often used to adapt models across domains, e.g. news to biomedical domain adaptation, or within one domain, e.g. WMT14 biomedical data to WMT18 biomedical data (Khan et al., 2018). Here, we apply transfer learning either across domains once or between domains iteratively. We obtain strong models that cover two disparate domains for both directions of the English-German language pair, and three related and overlapping domains for both directions of English-Spanish.

Transfer learning for domain adaptation involves using the performance of a model on some general domain A to improve performance on some other domain B: $A \rightarrow B$. However, if the two domains are sufficiently related, we suggest that task B could equally be used for transfer learning A: $B \rightarrow A$. The stronger general model A could then be used to achieve even better performance on other tasks: $B \rightarrow A \rightarrow B$, $B \rightarrow A \rightarrow C$, and so on. This is effectively a domain curriculum which concludes on the domain of interest.

Our WMT19 submission covers English-Spanish and English-German language pairs. For English-Spanish, we use the domain-labelled Scielo (Neves et al., 2016) dataset to provide two distinct domains, Health and Biological sciences ('Bio'), in addition to the complete biomedical dataset which includes both Scielo domains among other datasets ('All-biomed'). We therefore experiment with iterative transfer learning, in which a model adapted with transfer learning is finally tuned further on the original domain. For English-German we have only one large labelled biomedical-domain-relevant corpora, and so use standard fine-tuning from a general domain News model to a single in-domain All-biomed dataset.

4.2.2 Experimental setup

Data

We report on both translation directions for two language pairs: Spanish-English (es-en) and English-German (en-de). Table 4.1 lists the in- and out-of-domain data used to train

[4]https://ufal.mff.cuni.cz/ufal_medical_corpus

[5]Neves et al. (2016)

[6]https://github.com/biomedical-translation-corpora/medline (Jimeno Yepes et al., 2017)

[7]Dušek et al. (2017)

[8]http://statmt.org/wmt17/translation-task.html

[9]Sets: http://www.statmt.org/wmt19/translation-task.html. Filtering: Stahlberg, Saunders, de Gispert, et al. (2019)

[10]http://www.himl.eu/test-sets

	Domain	Training sets	Sentences pairs	Dev sets	Sentence pairs
es-en	All-biomed	UFAL Medical[4] Scielo[5] Medline titles[6] Medline abstracts Total filtered	639K 713K 288K 83K **1291K**	Khresmoi[7]	1.5K
	Health	Scielo health only Total filtered	587K **558K**	Scielo health	5K
	Bio	Scielo bio only Total filtered	126K **122K**	Scielo bio	4K
en-de	News	News corpus 2016-18[8] Paracrawl (filtered)[9]	92M 15M	Newstest-17[8]	3K
	All-biomed	UFAL Medical Medline abstracts Total filtered	2958K 33K **2156K**	Khresmoi Cochrane[10]	1.5K 467

Table 4.1 Training and validation data used in the WMT19 biomedical translation task. The English-German models were additionally pre-trained on very large general-domain datasets from the WMT19 news translation task. For both language pairs we use identical data when translating into and from English.

our biomedical domain evaluation systems. For en2de and de2en we additionally reuse strong general domain News models trained on data made available for the WMT19 news translation task, including filtered Paracrawl (Bañón et al., 2020). Details of data preparation and filtering for the News models are discussed more fully in Stahlberg, Saunders, de Gispert, et al. (2019).

For each language pair we use the same training data in both directions, and use a 32K-merge source-target BPE vocabulary (Sennrich, Haddow, and Birch, 2016d) trained on the 'base' domain training data (news for en-de, Scielo health for es-en)

We preprocess the data using Moses tokenization, punctuation normalization and truecasing. We then use a series of simple heuristics to filter the parallel datasets:

- Detected language filtering using the Python `langdetect` package[11]. In addition to mislabelled sentences, this step removes many sentences which are very short or have a high proportion of punctuation or HTML tags.
- Remove sentences containing more than 120 tokens or less than 3 tokens.
- Remove duplicate sentence pairs

[11] https://pypi.org/project/langdetect/

- Remove sentences where the ratio of source to target tokens is less than 1:3.5 or more than 3.5:1
- Remove pairs where more than 30% of either sentence is the same token.

Model, training and inference

We use the Tensor2Tensor (Vaswani, Bengio, et al., 2018) implementation of the Transformer model with the `transformer_big` setup for all NMT models. By default this model size limits batch size of 2K due to memory constraints. We delay gradient updates by a factor of 8, letting us effectively use a 16K batch size (Saunders, Stahlberg, de Gispert, et al., 2018). We train each domain model until it fails to improve on the in-domain validation set for three consecutive checkpoints, then perform checkpoint averaging over the final 10 checkpoints to obtain the final model (Junczys-Dowmunt et al., 2016).

At inference time we decode with beam size 4 using the SGNMT toolkit (Stahlberg, Hasler, Saunders, et al., 2017). For validation results we report cased BLEU scores with SacreBLEU (Post, 2018)[12]. These are the inference settings used throughout this thesis unless specified otherwise. Test results use case-insensitive BLEU to correspond to results released by the organizers.

4.2.3 WMT19 biomedical translation experiments

Iterative transfer learning improves over training on shuffled domains

	Transfer learning schedule	**es2en**			**en2es**		
		Khresmoi	**Health**	**Bio**	**Khresmoi**	**Health**	**Bio**
1	All-biomed	49.8	35.4	35.7	43.4	33.9	37.5
2	Health → All-biomed	**52.1**	36.7	37.0	44.2	35.0	39.0
3	Health	45.1	35.7	34.0	41.2	34.7	36.1
4	All-biomed → Health	48.9	36.4	35.9	43.0	35.2	38.0
5	Health → All-biomed → Health	51.1	**37.0**	37.2	44.0	**36.3**	39.5
6	Bio	37.4	29.3	35.8	36.0	30.1	39.5
7	All-biomed → Bio	48.0	34.6	37.2	43.2	34.1	40.5
8	Health → Bio	45.1	35.0	37.0	42.3	34.7	40.1
9	Health → All-biomed → Bio	50.6	36.0	**38.0**	**45.2**	35.3	**41.3**

Table 4.2 Development set BLEU for English-Spanish models with transfer learning. In each case transfer learning from another domain improves final performance on the relevant development set.

[12]SacreBLEU signature: `BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.3.2`

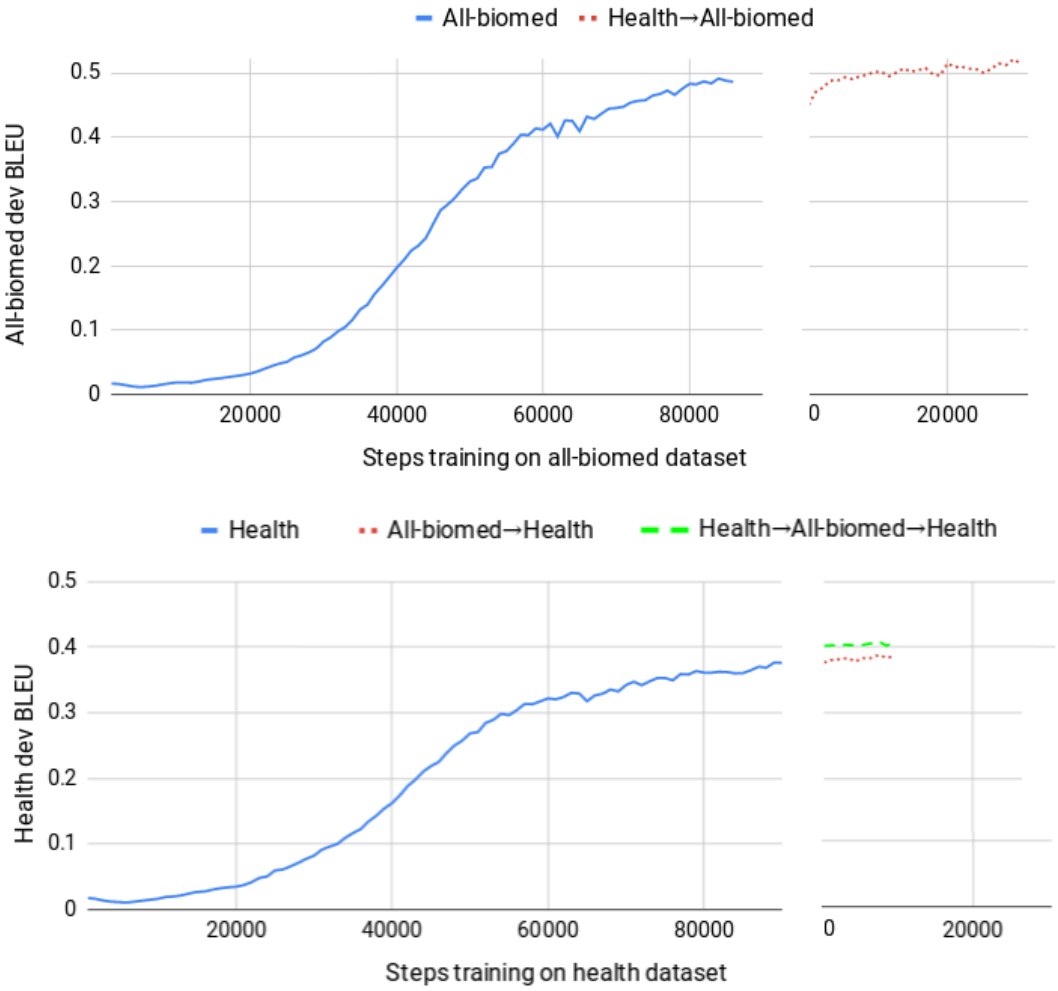


Fig. 4.1 Transfer learning for es2en domains. Top: standard transfer learning improves performance from a smaller (Health) to a larger (All-biomed) domain. Bottom: returning to the original domain after transfer learning provides further gains on Health.

Our iterative transfer learning experiments cover es2en and en2es to obtain models on three separate domains for evaluation. We use Health as the initial domain to train from scratch. We choose Health over All-biomed because it is significantly smaller and is more consistent in terms of topic and genre than the All-biomed set, which we expect would be harder to fit. We choose Health over Bio as it is four times larger and correspondingly likely to contain more varied language.

Once the Health model has converged on the Health validation set, we use it to initialize training on the larger, more diverse All-biomed corpus. When this transfer-learned All-biomed model has converged, we finally use it to initialize training on the Health data and Bio data separately for stronger models on those domains. Figure 4.1 shows the training

progression on Health and All-biomed validation sets. We also show training curves for the more typical transfer learning approach of training from scratch on All-biomed, which contains all other datasets, before fine-tuning on the narrower-domain Health data.

Table 4.2 gives single model validation scores for es2en and en2es models with standard and iterative transfer learning for various curricula that each end on the same domain. We find that the All-biomed domain gains 1-2 BLEU points when fine-tuned from the Health domain (line 2 vs line 1). Moreover, the Health and Bio domains benefit from iterative transfer learning (lines 5 and 9) relative to training from scratch (lines 3 and 6) and relative to standard transfer learning (lines 4, 7 and 8). These models benefit from a domain curriculum despite being trained more than once to convergence on the final domain in the iterative transfer learning case.

The results for the Health and Bio validation sets are worth highlighting in terms of the focus of this thesis on multi-domain performance. Why does All-biomed, the largest domain, not simply out-perform the other domains? All-biomed contains both the Health and Bio training data, and so has been trained on the same relevant training examples as the other models. However, the presence of other training data changes the All-biomed model's convergence. Results in line 1 on the domain-specific Health and Bio validation sets are even slightly weaker compared to models trained only on the Health (line 3) or Bio (line 6) subsets, let alone the stronger models that converge on those sets. This is despite all test and training domains under consideration being strongly related. This result illustrates that simply training a new model to convergence on all available data indiscriminately will not necessarily achieve the best performance on any specific sub-domain, even if such an approach was computationally practical.

Relatedly we notice some forgetting effects even with such strongly-related domains, or when fine-tuning on a subset of the pre-training domain. For example, in Table 4.2 the All-biomed $\rightarrow$ Health model (line 4) gains 1 BLEU on the Health validation set relative to the All-biomed model (line 1) that initializes it. However, it loses 0.9 BLEU relative to All-biomed on the more general Khresmoi set. We will discuss this forgetting effect and ways to mitigate it further in later chapters of this thesis.

Multi-domain ensembles perform well across domains

Table 4.3 gives validation and WMT19 biomedical test results for the models involved in the submission. Our first submission is the best All-biomed domain model, as the most topic-robust biomedical model, and our second is a uniform ensemble of all three models. By 'uniform' we mean that equal weighting is given to the predictions from each ensemble component during inference. Interestingly, the ensemble achieves approximately the same

	es2en				en2es			
	Khresmoi	**Health**	**Bio**	**Test**	**Khresmoi**	**Health**	**Bio**	**Test**
1: Health → All-biomed	52.1	36.7	37.0	42.4	44.2	35.0	39.0	44.9
1 → Health	51.1	**37.0**	37.2	-	44.0	**36.3**	39.5	-
1 → Bio	50.6	36.0	**38.0**	-	**45.2**	35.3	**41.3**	-
Uniform ensemble	**52.2**	36.9	37.9	**43.0**	45.1	35.6	40.2	**45.4**

Table 4.3 Validation and test BLEU for models involved in English-Spanish language pair submissions.

	de2en			en2de		
	Khresmoi	**Cochrane**	**Test**	**Khresmoi**	**Cochrane**	**Test**
News	43.8	46.8	-	30.4	40.7	-
News → All-biomed	44.5	47.6	27.4	31.1	39.5	26.5
Uniform ensemble	45.3	48.4	**28.6**	**32.6**	42.9	**27.2**

Table 4.4 Validation and test BLEU for models used in English-German language pair submissions.

result on each validation set as the best performing single model for es2en. For en2es the single Health and Bio models outperform the ensemble. For the test set, the ensemble improves by approximately 0.5 BLEU over the All-biomed model in both cases.

For English-German the initial and adapted domains – News and Biomedical respectively – are quite distinct. However, results in Table 4.4 show the strong initial out-of-domain model performing reasonably on in-domain data – less than 1 BLEU difference from the in-domain adapted model. This suggests that strength of general translation ability can in some sense make up for lack of domain-specific training examples. Moreover, a uniform ensemble of the two models gains 0.8 and 0.7 BLEU respectively over the in-domain model. This is a slightly clearer benefit than the 0.6 and 0.5 BLEU improvements from the English-Spanish results for three-model in-domain ensembles.

4.2.4 WMT19 biomedical translation task summary

Our WMT19 Biomedical submission covers the English-German and English-Spanish language pairs. For English-Spanish we use transfer learning iteratively to train single models which perform well on related but distinct domains. For English-German we adapt once to a relevant domain. In both cases we show further gains from multi-domain ensembles. However, simple transfer learning reduces performance on the pre-training domain even when that domain is very similar to the fine-tuning domain or is a superset of the fine-tuning data.

4.3 Genre-specific fine-tuning and the WMT20 biomedical translation task

In this section we discuss our participation in the WMT20 biomedical translation task[13]. We again submit translations for both directions of the English-German and English-Spanish language pairs. While the WMT19 submission involved training multiple models by transfer learning on related but distinct domains, we here describe a contrastive approach. We fine-tune existing strong models only on a single small, genre-matched adaptation set.

This is a popular approach for machine translation fixed-domain evaluations like the news and biomedical shared tasks. A common example is tuning on test sets released for the same shared task in previous years (Koehn, Duh, et al., 2018; Schamper et al., 2018; Stahlberg, Saunders, de Gispert, et al., 2019). It is a relatively efficient approach: adaptation sets might have only tens of thousands of sentence pairs, compared to millions of sentence pairs used to train the original model. The adaptation sets are also likely to contain sentences stylistically very similar to those in the test set.

For our submissions we start with the strong All-biomed models trained with iterative transfer learning for the WMT19 biomedical task (Tables 4.3 and 4.4), and adapt them further to Medline abstract training data (Bawden, Bretonnel Cohen, et al., 2019). This is a small and highly relevant training set, allowing extremely fast adaptation with the potential to let the NMT model adapt strongly to the domain of interest.

However, fine-tuning on relevant but small corpora has pitfalls. The small number of training examples exacerbates the effect of any noisy or poorly aligned sentence pairs. As well, overconfidence from over-fitting to a small, regular training set can result in poor translation hypotheses at test time. We use this section to highlight the potential benefits of small-dataset fine-tuning, as well as the potential risks of this type of exposure bias.

4.3.1 Small domain fine-tuning and exposure bias

Exposure bias for an autoregressive sequence decoder refers to a discrepancy between decoder conditioning during training and inference, reviewed in Sec. 2.3. Previous work has interpreted the risk of exposure bias primarily in terms of the model over-relying on correct gold target translations, resulting in error propagation when mistakes are made during inference. Here we focus instead on mistakes or misalignments in the training data which harm the model through teacher-forcing exposure. We suggest that exposure to this kind of imperfect training data can cause the model to make related mistakes during inference.

[13] http://www.statmt.org/wmt20/biomedical-translation-task.html

Sentences affected by exposure bias can be difficult to identify, particularly when the bias is partially caused by unreliable reference sentences. For this investigation we identify a specific feature of the Medline abstract training data which triggers noticeable translation errors after fine-tuning. The data contains instances in which either the source or target sentence contains the correct translation of the other sentence, but adds information that is not found in translation. For example, the following sentence appears in the English side of en-de Medline abstract training data:

[The effects of Omega-3 fatty acids in clinical medicine]. Effects of Omega-3 fatty acids (n-3 FA) in particular on the development of cardiovascular disease (CVD) are of major interest.

The aligned German sentence is:

Der Nutzen von Omega-3-Fettsäuren (n-3-FS) in der Medizin, hauptsächlich in der Prävention kardio- und zerebrovaskulärer Erkrankungen, wird aktuell intensiv diskutiert. (Translated: 'The uses of Omega-3 fatty acids in medicine, especially in prevention of cardiovascular and cerebrovascular diseases, are currently heavily discussed.')

Some of the English sentence is present in the German translation, but the square-bracketed article title is not. In this example it might be possible to remove only the segment in square brackets, but in other examples there is even less overlap, while source and target sentences may still be related and therefore challenging to filter. For example, the following English and German sentences also correspond with still less overlap:

[Conflict of interest with industry–a survey of nurses in the field of wound care in Germany, Australia and Switzerland]. Background.

Hintergrund: Pflegende werden zunehmend von der Industrie umworben. (Translated: 'Background: Nurses are being increasingly courted by industry.')

These examples are relatively frequent in Medline abstract data, especially in the form of titles. It is common to insert the English title of a non-English article into its translation, marked with square brackets (Patrias and Wendling, 2007). The title is however not present in the original non-English article[14]. This can cause models trained on English source sentences with titles to behave erratically when given sentences with square-bracketed titles at test time: an exposure bias effect.

One possible approach to this problem is simply removing tokens that trigger exposure bias from test sentences as a pre-processing step before inference. In the case of Medline abstracts, this means no square brackets for inference. Another approach is aggressively

[14]It is not unusual for human translators to add or discard information when translating (Darwish and Sayaheen, 2019; Puurtinen, 2003). This is a strong motivation for use of multiple human references when evaluating with BLEU, but such additional references are rarely available (Song, Cohn, et al., 2013).

filtering sentences which may be poorly aligned. However, with such a small training set, this risks losing valuable examples of domain-specific source and target language.

It is also important to note that square-bracketed title translations are not the only case of inexact training pairs, but are simply easily identifiable. Other less frequent examples might include unusual capitalization, or presence of non-translated words. One such case is in the first example given above, where 'cerebrovascular' is not included in the English sentence. The exact fine-tuning and test sets will determine specific exposure bias effects. Where the triggering feature is less obvious than title demarcation with square brackets, simple data-based techniques like filtering or inference-time pre-processing will be less effective.

We note that title translations are often removed from human references for the WMT biomedical task, precisely because they are often not well-aligned to the rest of the translated document. In these cases failure to translate the title will not negatively impact BLEU. However, we believe a biomedical translation model should be able to translate such sentences if required.

4.3.2 Experimental setup

Data

	Phase	**Training sets**	**Sentence pairs**	**Dev sets**	**Sentence pairs**
en-es	Pre-training	UFAL Medical	639K	Khresmoi	1.5K
		Scielo	713K		
		Medline titles	288K		
		Medline abstracts	83K		
		Total	**1291K**		
	Fine-tuning	Medline abstracts	**67.5K**	Biomedical19	825
en-de	Pre-training	UFAL Medical	2958K	Khresmoi	1.5K
		Medline abstracts	33K	Cochrane	467
		Total	**2156K**		
	Fine-tuning	Medline abstracts	**28.6K**	Biomedical19	808

Table 4.5 Biomedical training and validation data used in the WMT20 task (en-de models originally fine-tuned from News domain models as described in previous section). For both language pairs identical data was used in both directions. Bolded numbers are totals after filtering. Data sources are as for Table 4.1.

In this section we discuss the first part of our investigation into exposure bias connected to our WMT20 biomedical task submission. This consists of experiments adjusting data only, and results on validation data only. Our final submitted systems with results on the WMT20 test sets will be discussed in Chapter 5.

We report on two language pairs: English-Spanish (en-es) and English-German (en-de). Table 4.5 lists the data used to train our biomedical domain evaluation systems. BPE vocabularies and pre-training data are as for the WMT19 task (Sec. 4.2.2).

All of our approaches involve fine-tuning pre-trained models. We initialize fine-tuning with the All-biomed models obtained via iterative transfer learning (Tables 4.3 and 4.4). We fine-tune these models on Medline abstracts data. We validate on test sets from the 2019 Biomedical task, concatenating the source-target and target-source 2019 test sets for each language pair, and selecting only the 'OK' aligned sentences as determined by the organizers[15]. For each language pair we use the same fine-tuning data in both directions, and preprocess all data with Moses tokenization, punctuation normalization and truecasing.

Before fine-tuning we carry out detected language filtering on the Medline abstracts fine-tuning data using LangDetect, as for the 2019 task. However, for these training sets we find LangDetect has a tendency to incorrectly label short sentences or those with rare vocabulary (very common in Medline abstracts) as a random language. For each language pair we therefore filter out only sentences where LangDetect identifies the source sentence as belonging to the target language, and vice versa. We otherwise use the same filtering heuristics as for the WMT19 task (Sec. 4.2.2).

For the more aggressively-filtered 'no-title' experiments we additionally remove all lines containing multiple tokens in square brackets, which in medical writing are used to denote the English translation of a non-English article's title. This leaves 27.3K sentence pairs for en-de and 64.8K for en-es: about 96% of the filtered data in both cases.

Model, training and inference

Model architecture, training and inference procedure are as for the WMT19 task. For each approach we fine-tune on a single GPU, saving checkpoints every 1K updates, until fine-tuning validation set BLEU fails to improve for 3 consecutive checkpoints. Generally this took about 5K updates. We use a 4K effective batch size[16], which we found gave good performance on this small adaptation set. We then perform checkpoint averaging (Junczys-Dowmunt et al., 2016) over the final 3 checkpoints to obtain the final model.

We decode with beam size 4 using SGNMT. Validation scores are for case-*insensitive*, detokenized text obtained using SacreBLEU to correspond more closely to test Medline scores.

[15] Means of determining 'OK' sentences discussed in Bawden, Bretonnel Cohen, et al. (2019)

[16] 1K token batches with gradient updates delayed by a factor of 4

4.3.3 WMT20 biomedical translation experiments

Fine-tuning on a small, genre-matched dataset can lead to BLEU score gains

		de2en	en2de	es2en	en2es
1	Baseline	38.8	30.6	**48.5**	46.6
2	Fine-tuning from 1	40.9	**32.5**	**48.5**	46.0
3	Fine-tuning from 1, no-title	40.9	32.2	47.0	44.9
4	Checkpoint averaging 1	38.7	30.5	**48.5**	**47.1**
5	Checkpoint averaging 2	41.1	32.2	48.0	45.9
6	Checkpoint averaging 3	**41.4**	31.8	48.2	46.3

Table 4.6 Validation BLEU developing models used in English-German and English-Spanish language pair submissions. Scores for lines 1-3 are for the final individual checkpoint saved during fine-tuning on Medline abstracts data, with or without 'title' lines.

We first adapt the model to the full title-included Medline training set. Lines 1 and 2 in Table 4.6 show that small-domain fine-tuning can lead to over-fitting and reduced performance (en2es) but also potentially significant gains (en2de, de2en), depending on the quality of the original system. We note that the Medline abstracts training sets are also present in the All-biomed corpora used to train the baselines for these experiments. As in the previous section, tuning on a domain-specific subset of a previously-seen dataset has the potential to improve BLEU scores.

Checkpoint averaging (reviewed in Sec. 2.4.3) is a simple post-training scheme that often improves robustness to training on very variable datasets. We find that this procedure increased validation set BLEU for the en2es baseline, leaving validation BLEU performance mostly unchanged or decreasing in the other cases. The relative performance of baseline versus fine-tuned models is largely unchanged by checkpoint averaging.

Over-fitting to imperfect data can cause exposure bias effects

Even fine-tuning which increases BLEU scores can trigger exposure bias as a side-effect. Table 4.7 demonstrates the different behaviour of fine-tuned models when training with or without the mismatched sentences, or when performing inference without the triggering tokens. The baseline model gives reasonable translations, although the second translation example misses some vocabulary. Fine-tuning on the full abstracts set introduces exposure bias effects, even though BLEU on this set increases by 1.7 points.

In the first example, the fine-tuned hypothesis is completely unrelated to the source sentence. In the second example, the fine-tuned output is more plausible and therefore misleading. It misses the initial clause completely, and is phrased like the opening sentence

System	German hypothesis	Approximate English translation
English source	[Associations of work-related strain with subjective sleep quality and individual daytime sleepiness].	
Human translation	[Zusammenhang von arbeitsbezogenen psychischen Beanspruchungsfolgen mit subjektiver Schlafqualität und individueller Tagesschläfrigkeit.]	
Baseline	[Assoziationen arbeitsbedingter Belastung mit subjektiver Schlafqualität und individueller Tagesschläfrigkeit].	[Associations of work-related stress with subjective sleep quality and individual daytime sleepiness.]
Fine-tuning	*Zusammenfassung.*	*Summary*
No square brackets for inference	Assoziationen arbeitsbedingter Belastungen mit subjektiver Schlafqualität und individueller Tagesschläfrigkeit.	Associations of work-related stress with subjective sleep quality and individual daytime sleepiness.
Fine-tuning, no title data	[Assoziationen arbeitsbedingter Belastung mit subjektiver Schlafqualität und individueller Tagesschläfrigkeit].	[Associations of work-related stress with subjective sleep quality and individual daytime sleepiness].
English source	[Effectiveness of upper body compression garments under competitive conditions: a randomized crossover study with elite canoeists with an additional case study].	
Human translation	[Effektivität von Oberkörperkompressionsbekleidung unter Wettkampfbedingungen: eine randomisierte Crossover-Studie an Elite-Kanusportlern mit einer zusätzlichen Einzelfallanalyse.]	
Baseline	[Wirksamkeit von Oberkörperkompressionsbekleidungsstücken unter kompetitiven Bedingungen: Eine randomisierte Crossover-Studie mit Elite-Kanuten mit einer *Additionsstudie*].	Effectiveness of upper body compression garments under competitive conditions: a randomized crossover study with elite canoeists with an *addition study*.
Fine-tuning	Eine randomisierte Crossover-Studie mit Elite-Kanuten mit einer *Additional Case Study wurde durchgeführt.*	A randomized crossover study with elite canoeists with an additional case study *was carried out*.
No square brackets at inference	Effektivität von Oberkörper-Kompressionsbekleidungsstücken unter kompetitiven Bedingungen: Eine randomisierte Crossover-Studie mit Spitzenkanuten mit einer zusätzlichen Fallstudie.	Effectiveness of upper body compression garments under competitive conditions: a randomized crossover study with leading canoeists with an additional case study.
Fine-tuning, no title data	[Effektivität von Oberkörperkompressionsbekleidungsstücken unter kompetitiven Bedingungen: Eine randomisierte Crossover-Studie mit Elite-Kanuten mit einer *Additional Case Study*].	[Effectiveness of upper body compression garments under competitive conditions: a randomized crossover study with elite canoeists with an additional case study]

Table 4.7 Two sentences from the English-German 2020 test set with hypothesis translations from various models (title casing removed for clarity). Examples demonstrate the effects of exposure bias from fine-tuning on imperfectly aligned training sentences. Notable hypothesis departures from the reference are *emphasized*.

of a paper rather than as a title. It also features the untranslated phrase 'Additional Case Study'.

We suggest two data-based schemes for coping with this. The first is inference-time preprocessing to remove triggering (square-bracket) tokens. The second fine-tunes instead on a 'no-title' version of the adaptation set with misaligned title sentence pairs aggressively filtered away. As can be seen in Table 4.7, both approaches are effective at reducing exposure bias effects.

We find that preprocessing to remove square brackets leaves other sentences unchanged, and has negligible effect on BLEU score since the affected sentences rarely have human references. By contrast, a model fine-tuned on the filtered data must translate all sentence pairs, and can impact overall BLEU either positively or negatively. We therefore also report validation BLEU with this model in lines 3 and 6 of Table 4.6.

Fine-tuning on filtered data gives slightly better results than fine-tuning on unfiltered data for de2en with checkpoint averaging, as can be seen in line 6 of Table 4.6. Since the added information in 'title' sentences is on the English side, this suggests that training on a target sentence with extra information may harm translation.

However, filtering these sentences results in performance degradation as measured by BLEU for the en2de model. This result suggests that these can be valuable training examples when the extra information is on the source side. In other words, source side additive noise may improve NMT model robustness. By contrast removing the information – source side noise in the form of deletions – harms performance. We note that prior work on back translation for NMT similarly finds that applying noise to source sentences improves translation performance. However such work has found that deletions are also useful for noising back translation, perhaps because back translation may tend towards over-generation (Edunov et al., 2018b).

4.3.4 WMT20 biomedical translation task summary

Our WMT20 Biomedical submission investigates improvements on the English-German and English-Spanish language pairs under a single strong model. In particular, we focus on the behaviour of models trained on sentences with some predictable irregularities. We find that aggressively filtering target sentences can help overall performance, but that aggressively filtering source sentence tends to hurt performance. We also highlight that such data-centric approaches to exposure bias effects are dependent on knowledge of the 'problem' sentences, and are therefore not applicable in all cases.

4.4 Conclusions

This chapter presents data-centric approaches to changing NMT model behaviour. Our examples revolve around selecting data labelled as domain relevant in terms of provenance, topic and/or genre, then adapting a model by means of some determined domain curriculum. These techniques lead to strongly performing individual systems used in submissions to WMT biomedical evaluation campaigns. Further development of the final models used in these submissions as well as analysis of the evaluation results will be discussed in Sec. 5.3.3 and 6.3.2.

Domain adaptation via model fine-tuning has been heralded as a simple and easy solution to the problem of unseen data or a new domain (Federico, 2018). However, such approaches can themselves introduce potentially serious performance degradation. We have therefore also highlighted the disadvantages of these purely data-centric approaches to domain adaptation. One disadvantage is forgetting: a model adapted to one domain will experience performance degradation on domains it previously translated well. Another disadvantage is the risk of domain mismatch and exposure bias, particularly when fine-tuning on small or easily over-fitted datasets. This can be addressed with data-centric techniques such as careful filtering or preprocessing, but such schemes require foreknowledge of the problematic sentence pairs. We will explore alternative options in Chapters 5 and 6.

Chapter 5

Training schemes to mitigate side-effects of NMT domain adaptation

This chapter draws from the following publications: Saunders, Stahlberg, de Gispert, et al. (2019) in Sec. 5.2, and Saunders, Stahlberg, and Byrne (2020) and Saunders and Byrne (2020a) in Sec. 5.3. Some results in Sec. 5.2 are from my contributions to Stahlberg, Saunders, de Gispert, et al. (2019)

5.1 Motivation

Neural Machine Translation (NMT) models often reach good performance on their training domain. However, the domain of sentences presented at inference time may differ from any training domain data. In this case of domain mismatch even models with strong performance on a broad domain can translate poorly. As explored in Chapter 4, data-centric approaches to this problem generally involve selecting some additional dataset related to the domain of interest and using the new data to fine-tune the model. However, purely data-centric approaches to domain adaptation can lead to new difficulties. In this chapter we focus on two problems in particular.

The first problem, 'catastrophic forgetting', occurs when translating the original domain. If a model with strong performance on domain A is fine-tuned on domain B, it often gives strong translation performance on domain B at the expense of extreme performance degradation on domain A. This is especially problematic when the domain of the test data is not known or may change over time. Ideally, a single model would be capable of translating text from multiple domains, regardless of when during training a given domain was learned.

The second problem, 'exposure bias', occurs when translating the new domain. It is particularly common when the fine-tuning dataset is small or very regular but still contains small irregularities such as misaligned sentences or idiosyncratic vocabulary use. This can cause problematic behaviour when the model is presented with new test data. In Sec. 4.3, we presented an example where almost every source sentence containing text in square brackets corresponded to a target sentence containing broadly unrelated text. The model then translated test sentences with square brackets also into unrelated text. Exposure bias is particularly relevant in cases of domain mismatch, where the test sentence domain is distinct from the fine-tuning domain (Wang and Sennrich, 2020).

In this chapter we explore variations on fine-tuning procedures for NMT domain adaptation. We focus on adaptation schemes that can address these two problems without varying the adaptation dataset itself. Our aim is to address the second research question raised in Sec. 1.1.1 by investigating adaptation schemes that make good use of a given adaptation set while avoiding the negative side-effects of fine-tuning. We also touch on the third research question: robustness to unknown or mismatched domain test sentences.

In Sec. 5.2 we apply various regularized adaptation objectives during domain adaptation to address the catastrophic forgetting problem and to allow multi-domain adaptation. In Sec. 5.3 we develop a robust form of discriminative training, doc-MRT. Doc-MRT allows performance improvements during model fine-tuning, and is effective at reducing exposure bias effects in fine-tuning for the previously discussed WMT20 biomedical task.

5.2 Regularized adaptation: addressing the 'catastrophic forgetting' problem

This section describes our exploration of parameter regularization schemes for NMT domain adaptation. Our aim is to allow translation of new domains while preserving performance on previously learned domains. Parameter regularization achieves this by holding model parameters close to their pre-trained values. This requires storing the pre-trained parameters during adaptation, but has no computational impact during inference and potentially allows minimal or no catastrophic forgetting of previously learned domains.

Several parameter regularization schemes have previously been applied to NMT domain adaptation, as reviewed in Sec. 3.4.2. In this section we focus on two related schemes: L2 regularization and EWC regularization. We include comparisons to straightforward non-regularized fine-tuning, as well as checkpoint averaging (reviewed in Sec. 2.4.3). Unlike regularized fine-tuning, checkpoint averaging is applied after training and is domain-agnostic.

However, it has a similar effect to regularized fine-tuning, in that it tends to 'smooth' any parameter fluctuations or variation from earlier versions of the model included in averaging.

We explore forgetting across both related domains and very distinct domains for English-to-German and English-to-Spanish translation, including the case of adapting sequentially to three domains, and find that EWC outperforms L2 regularization. We find that applying EWC regularization only to model embeddings can still significantly reduce forgetting. We finally show that EWC can allow translation improvements when fine-tuning a strong model on a small, trusted dataset which has the same domain as the original model.

5.2.1 L2 Regularization and Elastic Weight Consolidation

We briefly re-introduce regularized fine-tuning for domain adaptation with the terminology and implementation choices used in our experiments. In all cases we assume an NMT model is first trained to convergence on some pre-training task PT. Importantly, this does not necessarily mean training on a single corpus. PT can cover multiple domains, whether via sequential adaptation or by adapting to mixed datasets.

During domain adaptation, pre-trained parameters θ^{PT} are fine-tuned on some new domain. Without regularization, catastrophic forgetting can occur: performance degradation on domains only covered by pre-training as parameters adjust to the new objective. A regularized objective is:

$$L(\theta) = L_{CE}(\theta) + \Lambda \sum_j F_j(\theta_j - \theta_j^{PT})^2 \tag{5.1}$$

where $L_{CE}(\theta)$ is the cross-entropy loss when training on the new task. In this section we compare three cases:

- **No-reg**, where $\Lambda = 0$
- **L2**, where $F_j = 1$ for each parameter index j
- **EWC**, where $F_j = \mathbb{E}\left[\nabla^2 L_{CE}(\theta_j^{PT})\right]$, a sample estimate of task PT Fisher information. This effectively measures the importance of θ_j to pre-training task PT.

For L2 and EWC we tune Λ on the validation sets for new and pre-training tasks to balance forgetting against new-domain performance.

5.2.2 Experimental setup

Data

We report on Spanish-to-English (es-en) and English-to-German (en-de) translation. For es-en we use the Scielo corpus (Neves et al., 2016), with Health as the general domain, adapting to Biological Sciences ('Bio'). We hold out 1K randomly selected sentence pairs from each labelled training corpus to act as a validation set. We evaluate on the domain-labeled Health and Bio 2016 test data.

The en-de general domain is News. We train on data made available for the 2018 WMT news translation shared task (Bojar, Federmann, et al., 2018), with all data except ParaCrawl oversampled by 2 (Sennrich, Birch, et al., 2017). We validate on the WMT news task WMT17 test set and evaluate on WMT18. We adapt first to the IWSLT TED talks task, validating on the 2015 test set and evaluating on the 2016 test set (Cettolo, Niehues, Stüker, Bentivogli, Cattoni, et al., 2016), and then sequentially to the WMT IT corpus, using the provided dev and test set from the 2017 APE task (Turchi et al., 2017).

We filter training sentences for minimum three tokens and maximum 120 tokens, and remove sentence pairs with length ratios higher than 4.5:1 or lower than 1:4.5. Table 5.1 shows filtered training sentence counts. Each language pair uses a joint source-target 32K-merge BPE vocabulary trained on the general domain (Sennrich, Haddow, and Birch, 2016d).

Language pair	Domain	Training	Dev	Test
es-en	Health	586K	1K	5K
	Bio	125K	1K	4K
en-de	News	22.1M	3K	3K
	TED	146K	1.1K	1.1K
	IT	11K	1K	2K

Table 5.1 Corpora sentence pair counts

Model, training and inference

We use the Tensor2Tensor (Vaswani, Bengio, et al., 2018) implementation of the Transformer model with the `transformer_base` setup for all NMT models with a 4K token batch size. We note that the different hyperparameter choices[1] result in different scores compared to baseline models on the same language pairs and domains in the previous chapter.

[1]The `transformer_base` model has embedding size 512 and 8 attention heads, while `transformer_big` has embedding size 1024 and 16 heads. We use smaller models for speed and storage efficiency when not preparing systems for evaluation campaigns.

We determine regularization weight Λ for L2 and EWC by tuning on the validation sets. When adapting via EWC sequentially to a third en-de domain, we re-estimate the Fisher information over data from both pre-training domains. We estimate the Fisher information for EWC with pre-training model parameters frozen. The data used for estimation is a random sample of 1K mini-batches from the es-en Health training dataset, and over 5K mini-batches for the much larger en-de News and News + TED training datasets.

We adapt for the same number of mini-batch updates with all fine-tuning schemes. For results with checkpoint averaging we obtain the final model by averaging over the final 10 checkpoints. At inference time we decode with beam size 4 in SGNMT and evaluate with case-sensitive detokenized BLEU using SacreBLEU.

5.2.3 Regularized adaptation experiments

Checkpoint averaging is complementary to regularized fine-tuning

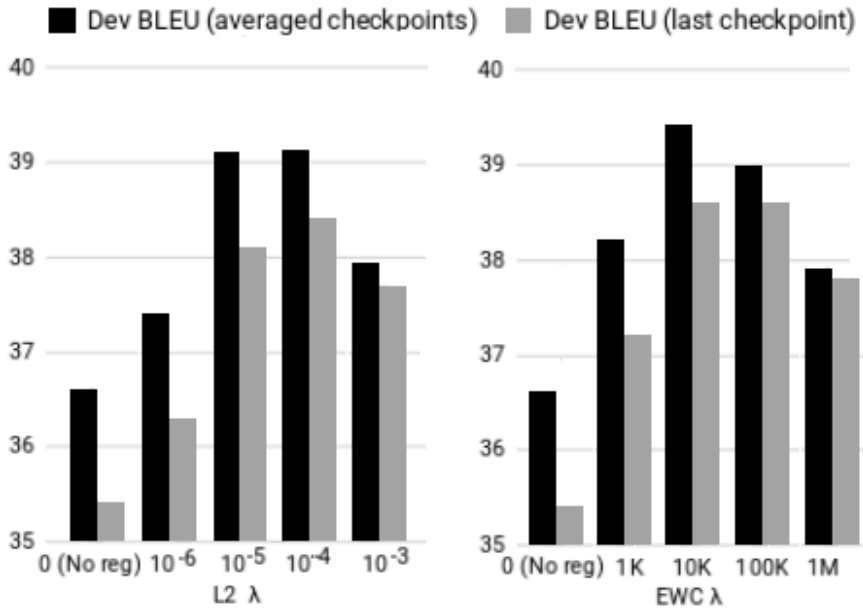


Fig. 5.1 Combined Health + Bio validation set BLEU when tuning Λ for es-en

We first describe the process for setting regularization weight Λ (Eq. 5.1). This is required for both EWC and L2. As part of this experiment, we compare validation results with and without checkpoint averaging. Figure 5.1 shows an example of tuning for es-en.

We tune on combined Health and Bio validation BLEU for inference with the final checkpoint and with averaged checkpoints over the final 10K training steps. Although we

select for the best combined result over domains, Λ can also be adjusted to for better new domain or pre-training domain performance as desired.

Model checkpoints from earlier in fine-tuning experience less forgetting, so averaging earlier and later checkpoints may give additional regularization. Figure 5.1 indicates that this is indeed the case. Checkpoint averaging gives a small regularization effect compared to L2 and EWC, and affects both schemes similarly. We report our following results with checkpoint averaging, except when adapting to the IT dataset where fine-tuning is extremely brief.

EWC gives less forgetting and better in-domain performance than L2 regularization

	Training scheme	Health	Bio
1	Health	**35.9**	33.1
2	Bio	29.6	36.1
3	Health and Bio	35.8	37.2
4	1 then Bio, No-reg	30.3	36.6
5	1 then Bio, L2	35.1	37.3
6	1 then Bio, EWC	35.2	**37.8**

Table 5.2 Test BLEU for es-en adaptive training. EWC reduces forgetting compared to other fine-tuning methods, while offering the greatest improvement on the new domain.

	Training scheme	News	TED	IT
1	News	37.8	25.3	35.3
2	TED	23.7	24.1	14.4
3	IT	1.6	1.8	39.6
4	News and TED	38.2	25.5	35.4
5	1 then TED, No-reg	30.6	**27.0**	22.1
6	1 then TED, L2	37.9	26.7	31.8
7	1 then TED, EWC	**38.3**	**27.0**	33.1
8	5 then IT, No-reg	8.0	6.9	56.3
9	6 then IT, L2	32.3	22.6	56.9
10	7 then IT, EWC	35.8	24.6	**57.0**

Table 5.3 Test BLEU for en-de adaptive training, with sequential adaptation to a third task. EWC-tuned models give the best performance on each domain.

We wish to improve performance on new domains without reduced performance on the general domain. For es-en results in Table 5.2, the Health and Bio tasks overlap, but catastrophic forgetting still occurs under no-reg (line 3). Regularization reduces forgetting and allows further improvements on Bio over unregularized fine-tuning. We find EWC (line

6) outperforms the L2 approach proposed for NMT adaptation by Barone et al. (2017) (line 5), both in terms of learning the new task and for reducing forgetting.

In the en-de News/TED task (Table 5.3), all fine-tuning schemes give similar improvements on TED. One notable difference is the relative performance on the pre-training News domain. EWC (line 7) outperforms both no-reg (line 5) and L2 (line 6) on News, not only reducing forgetting but giving 0.5 BLEU improvement over the baseline News model. Using EWC for domain adaptation may therefore be particularly advantageous when the original and fine-tuning domains are similar.

The IT dataset is very small: training on IT data alone results in over-fitting, with a 17 BLEU improvement as well as forgetting of the previous tasks under no-reg fine-tuning (line 8). EWC (line 10) reduces forgetting on two previous tasks while further improving on the target domain.

EWC can still perform well if only applied to embeddings

For every parameter θ_j regularized by EWC, the model must store values for θ_j^{PT} and F_j. Reducing the number of regularized parameters is therefore a question of practical interest. Also of practical interest is convergence rate under EWC. All regularization schemes in Table 5.3 are adapted for the same number of epochs over each dataset, but we are also interested in the performance of EWC when adaptation time is limited.

We focus on News-to-TED tuning as in other cases some level of forgetting occurs even with full EWC regularization. By contrast, for News to TED adaptation scores improve on both domains. The News test set score increases by 0.5 BLEU after fine-tuning on TED with EWC regularization, despite the same score decreasing by over 7 BLEU after fine-tuning with no regularization.

Table 5.4 gives News and TED test BLEU scores when applying EWC regularization to different model subnetworks. Models are adapted for fewer steps than in Table 5.3, allowing us to explore relative convergence rates. Results with no-reg adaptation are approximately unchanged on TED, and results with EWC applied to all parameters are unchanged on News. However, as we might expect, shorter no-reg fine-tuning means less forgetting of News, and shorter EWC (all parameters) adaptation means lower scores on the new domain.

Interestingly, it appears that good results on both domains are possible even with a shorter adaptation period by applying EWC only to embedding parameters, with other parameters able to vary without regularization. Applying EWC only to the encoder or decoder has almost no effect on catastrophic forgetting. However, the best results in terms of reduced forgetting still come when EWC is applied to all parameters, even in the case of longer adaptation given in Table 5.3.

EWC-regularized parameters	News	TED
Baseline	37.8	25.3
None (No-reg)	34.4	**27.2**
All	**38.3**	26.6
Encoder only	34.4	26.9
Decoder only	34.6	26.8
Embeddings only	38.0	**27.2**

Table 5.4 Test BLEU for en-de adaptation from the News domain to the the TED domain, applying EWC regularization only to subsets of the Transformer model parameters. All other parameters vary freely. These models are adapted to TED for fewer steps than models in Table 5.3 to highlight the effect of EWC on convergence rate, resulting in slightly different scores for the all-EWC and no-reg models.

WMT19 news task: EWC improves performance when tuning on validation sets

An effective way to improve an already strong model is fine-tuning on small datasets that are known to be very similar to the test data, as discussed in Chapter 4. As shown for News-to-TED task adaptation in Table 5.3, EWC can give performance gains over simple fine-tuning when the domains are related. We explore this possibility as part of our submission to the WMT 2019 News translation task.

We fine-tune strong English-German NMT models on 2008-2016 WMT News task test sets. Since memory is not a limiting factor and we are most interested in avoiding over-fitting, we apply EWC regularization to all model parameters. We tune Λ for EWC on newstest-2017 and use newstest-2018 as a test set. Details of model and data preparation are given in Stahlberg, Saunders, de Gispert, et al. (2019). To avoid over-fitting, we fine-tune for 1K-2K iterations (determined by tuning set performance), saving checkpoints every 500 iterations. We average checkpoints before validation. The case-sensitive BLEU scores for English-to-German and German-to-English NMT are given in Table 5.5.

Fine-tuning	Checkpoint averaging	En-De	De-En
None	×	46.7	46.5
None	✓	46.6	46.4
No-reg	×	47.1	46.6
No-reg	✓	47.3	**46.8**
EWC	×	47.1	46.4
EWC	✓	**47.8**	**46.8**

Table 5.5 BLEU on newstest-2018 when fine-tuning large English-German models on past WMT test sets without regularization and with EWC regularization. EWC is complementary to checkpoint averaging.

Averaging the last few unadapted checkpoints from regular training does not improve performance. Averaging all fine-tuning checkpoints with the final unadapted checkpoint gives small improvements over no-reg fine-tuning without averaging. The best results come when combining EWC regularization during training with checkpoint averaging after training. As in Fig 5.1 we find that checkpoint averaging is complementary to EWC.

5.2.4 Regularized adaptation summary

We report on training techniques that adapt NMT to new domains while preserving performance on the original domain. We demonstrate that EWC effectively regularizes NMT fine-tuning. We find that EWC outperforms other training regularization schemes reported for NMT at reducing forgetting and at improving performance on the new domain. We also find EWC is complementary to checkpoint averaging, which is effectively a post-training, pre-inference parameter regularization scheme.

5.3 Using context in MRT objectives for robustness

So far, approaches to adaptation in this thesis have used the standard cross-entropy loss function. In this section we explore variations on Minimum Risk Training (MRT), a discriminative loss function. As reviewed in Sec. 2.3.1, the MRT training objective is of interest to NMT generally. It is of relevance to this thesis in particular for two main reasons:

1. In the NMT literature, to the best of our knowledge, MRT is exclusively applied to fine-tune a model that has already converged under a maximum likelihood objective. MRT therefore fits naturally into an exploration of improvements to NMT models via fine-tuning and adaptation.

2. There is some indication that MRT may be effective at reducing the effects of exposure bias (Wang and Sennrich, 2020). As previously discussed in Sec. 4.3.1, exposure bias can be a particular difficulty where there is a risk of over-fitting a small dataset, which is often the case for domain adaptation.

MRT as applied in the NMT literature exclusively uses sequence-level objectives as the cost function (Sec. 2.3.1). A typical sequence objective is based on sentence-level BLEU (sBLEU, reviewed in Sec 2.4.4). However sBLEU, even if aggregated over sentences, is only an approximation of the metric actually used for machine translation evaluation, which is document-level BLEU. Beyond translation, many metrics for natural language tasks do not have robust sentence-level approximations.

A logical progression is the extension of sequence-level NMT training objectives to include context from outside the sentence. However, to the best of our knowledge, no attempt has previously been made to extend sequence-level neural training objectives to include document-level reward functions. This is despite document-level BLEU being arguably the most common NMT metric, and being the function originally optimized by Minimum Error Rate Training (MERT) for Statistical Machine Translation (SMT) (Och, 2003).

In this section we present a document-level approach to sequence-level objectives which brings the training objective closer to the actual evaluation metric, using MRT as a representative example. We refer to our scheme as doc-MRT, by way of contrast with the standard MRT formulation which uses a sequence-level objective, which we here refer to as seq-MRT. We demonstrate doc-MRT under document-level BLEU as well as Translation Edit Rate (TER) (Snover, 2006). We experiment both with pseudo-documents where sentences are assigned randomly to a mini-batch, and true document context where all sentences in the batch are from the same document.

We also apply our scheme to supervised Grammatical Error Correction (GEC). Use of neural models for GEC is increasingly popular (Sakaguchi et al., 2017; Stahlberg, Bryant, et al., 2019; Xie et al., 2016). We show gains in GEC metrics GLEU (Napoles, Sakaguchi, Post, et al., 2015) and M2 (Dahlmeier and Ng, 2012).

Finally, we return to the problem of exposure bias when fine-tuning on small, imperfect biomedical training sets (Sec. 4.3.1). We show that doc-MRT fine-tuning is effective at improving performance in terms of BLEU score. It also reduces the degenerative effects of exposure bias without requiring knowledge of specific triggering sentences or tokens, unlike the data-centric schemes described in the previous chapter.

5.3.1 Document-level MRT

A risk gradient view of sequence-level MRT

Sentence-level MRT for NMT aims to minimize the expected loss on training data with a loss function between sampled target sentences $\boldsymbol{y}$ and corresponding reference sentences $\boldsymbol{y}^*$. A sequence-level loss function between sample and reference is defined as $\Delta(\boldsymbol{y}, \boldsymbol{y}^*)$, which may be non-differentiable. For NMT a common cost function is $\Delta(\boldsymbol{y}, \boldsymbol{y}^*)$ = 1 - sBLEU$(\boldsymbol{y}, \boldsymbol{y}^*)$, where sBLEU is smoothed by setting initial n-gram counts to 1 (Edunov et al., 2018a).

Consider taking N samples for each of the S sentences in a mini-batch. We write the cost function between the s^{th} reference in a mini-batch, $\boldsymbol{y}^{(s)*}$, and its n^{th} sample, $\boldsymbol{y}_n^{(s)}$, as $\Delta(\boldsymbol{y}_n^{(s)}, \boldsymbol{y}^{(s)*})$.

The seq-MRT loss function to be minimized is as follows:

$$\begin{aligned} R(\theta) &= \sum_{s=1}^{S} \mathbb{E}_{y^{(s)}|x^{(s)};\theta,\alpha} \left[\Delta(y^{(s)}, y^{*(s)}) \right] \\ &= \sum_{s=1}^{S} \sum_{n=1}^{N} \frac{P(y_n^{(s)}|x^{(s)};\theta)^{\alpha}}{\sum_{n'=1}^{N} P(y_{n'}^{(s)}|x^{(s)};\theta)^{\alpha}} \Delta(y_n^{(s)}, y^{*(s)}) \\ &= \sum_{s=1}^{S} \sum_{n=1}^{N} Q(y_n^{(s)}|x^{(s)};\theta,\alpha) \Delta(y_n^{(s)}, y^{*(s)}) \end{aligned} \tag{5.2}$$

We can write the gradient of the log Q function with respect to a specific parameter θ_i:

$$\begin{aligned} &\frac{\partial}{\partial \theta_i} \log Q(y_n^{(s)}|x^{(s)};\theta,\alpha) \\ &= \frac{\partial}{\partial \theta_i} \alpha \log P(y_n^{(s)}|x^{(s)};\theta) - \frac{\partial}{\partial \theta_i} \log \sum_{n'=1}^{N} P(y_{n'}^{(s)}|x^{(s)};\theta)^{\alpha} \\ &= \alpha \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) - \alpha \frac{\sum_{n'=1}^{N} P(y_{n'}^{(s)}|x^{(s)};\theta)^{\alpha-1} \frac{\partial}{\partial \theta_i} P(y_{n'}^{(s)}|x^{(s)};\theta)}{\sum_{n'=1}^{N} P(y_{n'}^{(s)}|x^{(s)};\theta)^{\alpha}} \\ &= \alpha \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) - \alpha \frac{\sum_{n'=1}^{N} P(y_{n'}^{(s)}|x^{(s)};\theta)^{\alpha} \frac{\partial P(y_{n'}^{(s)}|x^{(s)};\theta)/\partial \theta_i}{P(y_{n'}^{(s)}|x^{(s)};\theta)}}{\sum_{n'=1}^{N} P(y_{n'}^{(s)}|x^{(s)};\theta)^{\alpha}} \\ &= \alpha \Big(\frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) - \mathbb{E}_{y^{(s)}|x^{(s)};\theta,\alpha} \left[\frac{\partial}{\partial \theta_i} \log P(y^{(s)}|x^{(s)};\theta) \right] \Big) \end{aligned} \tag{5.3}$$

Given a sampled space of N sequences, and writing $\mathbb{E}_{y^{(s)}|x^{(s)};\theta,\alpha} = \mathbb{E}_{z^{(s)}}$ and $\Delta(y^{(s)}, y^{*(s)}) = \Delta_n^{(s)}$ for brevity, we take the partial derivative of the objective $R(\theta)$ with respect to a model parameter θ_i:

$$\begin{aligned} \frac{\partial R(\theta)}{\partial \theta_i} &= \sum_{s=1}^{S} \frac{\partial}{\partial \theta_i} \sum_{n=1}^{N} \Delta_n^{(s)} Q(y_n^{(s)}|x^{(s)};\theta,\alpha) \\ &= \sum_{s=1}^{S} \sum_{n=1}^{N} Q(y_n^{(s)}|x^{(s)};\theta,\alpha) \Delta_n^{(s)} \frac{\partial}{\partial \theta_i} \log Q(y_n^{(s)}|x^{(s)};\theta,\alpha) \\ &= \sum_{s=1}^{S} \mathbb{E}_{z^{(s)}} \left[\Delta_n^{(s)} \frac{\partial}{\partial \theta_i} \log Q(y^{(s)}|x^{(s)};\theta,\alpha) \right] \end{aligned} \tag{5.4}$$

Using the gradient of the log Q function from Eq. 5.3:

$$\begin{aligned}\frac{\partial R(\theta)}{\partial \theta_i} &= \sum_{s=1}^{S} \mathbb{E}_{z^{(s)}} \left[\Delta_n^{(s)} \alpha \Big(\frac{\partial}{\partial \theta_i} \log P(y^{(s)}|x^{(s)};\theta) - \mathbb{E}_{z^{(s)}} \left[\frac{\partial}{\partial \theta_i} \log P(y^{(s)}|x^{(s)};\theta) \right] \Big) \right] \\ &= \sum_{s=1}^{S} \alpha \Big(\mathbb{E}_{z^{(s)}} \left[\Delta_n^{(s)} \frac{\partial}{\partial \theta_i} \log P(y^{(s)}|x^{(s)};\theta) \right] - \mathbb{E}_{z^{(s)}} \left[\Delta_n^{(s)} \right] \mathbb{E}_{z^{(s)}} \left[\frac{\partial}{\partial \theta_i} \log P(y^{(s)}|x^{(s)};\theta) \right] \Big)\end{aligned} \tag{5.5}$$

The final line of Eq. 5.5 corresponds to Equation 14 from Shen et al. (2016). A variance-reduced unbiased estimate of the gradient suggested by Shannon (2017) involves averaging over samples in a way related to the Reinforce algorithm (Williams, 1992). This is done via a Monte Carlo approximation for (e.g.) the expected cost $\mathbb{E}_{y^{(s)}|x^{(s)};\theta,\alpha}[\Delta_n^{(s)}]$:

$$\mathbb{E}_{y^{(s)}|x^{(s)};\theta,\alpha} \left[\Delta_n^{(s)} \right] = \overline{\Delta_n^{(s)}} = \frac{1}{N} \sum_{n=1}^{N} \Delta_n^{(s)} \tag{5.6}$$

An estimate of the loss gradient, from Eq. 5.5, is then:

$$\begin{aligned}\frac{\partial R(\theta)}{\partial \theta_i} &\approx \sum_{s=1}^{S} \alpha \Big(\frac{1}{N} \sum_{n=1}^{N} \big(\Delta_n^{(s)} \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) \big) - \frac{1}{N} \sum_{n=1}^{N} \Delta_n^{(s)} \frac{1}{N} \sum_{n=1}^{N} \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) \Big) \\ &= \sum_{s=1}^{S} \alpha \Big(\frac{1}{N} \sum_{n=1}^{N} \big(\Delta_n^{(s)} \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) \big) - \overline{\Delta_n^{(s)}} \frac{1}{N} \sum_{n=1}^{N} \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) \Big) \\ &= \sum_{s=1}^{S} \alpha \Big(\frac{1}{N} \sum_{n=1}^{N} \big(\Delta_n^{(s)} - \overline{\Delta_n^{(s)}} \big) \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) \Big)\end{aligned} \tag{5.7}$$

The risk gradient can be approximated with variance reduction:

$$\frac{\partial R(\theta)}{\partial \theta_i} \approx \sum_{s=1}^{S} \frac{\alpha}{N-1} \sum_{n=1}^{N} (\Delta_n^{(s)} - \overline{\Delta_n^{(s)}}) \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) \tag{5.8}$$

Document-level MRT

By analogy with sequence-level MRT, we can now consider doc-MRT over batches of S sentence pairs, which we treat as a pseudo-document.

Let $X = [x^{(1)},\ldots,x^{(S)}]$ be the source document, $Y = [y^{(1)},\ldots,y^{(S)}]$ be a document of candidate translations, and $Y^* = [y^{*(1)},\ldots,y^{*(S)}]$ be the reference translations. Document-level metric $D(Y,Y^*)$, which may be non-differentiable, replaces the sequence-level metric

$\Delta(y^{(s)}, y^{*(s)})$. We define the document-level risk:

$$\begin{aligned} R(\theta) &= \mathbb{E}_{Y|X;\theta,\alpha}\left[D(Y,Y^*)\right] \\ &= \sum_Y \frac{P(Y|X;\theta)^\alpha}{\sum_{Y'} P(Y'|X;\theta)^\alpha} D(Y,Y^*) \\ &= \sum_Y Q(Y|X;\theta,\alpha) D(Y,Y^*) \end{aligned} \tag{5.9}$$

By analogy with the derivation from Eq. 5.2 to Eq. 5.8, making the same Monte Carlo assumptions, and with N now as the number of sampled documents Y, the loss gradient becomes:

$$\frac{\partial R(\theta)}{\partial \theta_i} \approx \frac{\alpha}{N-1} \sum_Y (D(Y,Y^*) - \overline{D(Y,Y^*)}) \frac{\partial}{\partial \theta_i} \log P(Y|X;\theta) \tag{5.10}$$

If a sample for each sentence $y^{(s)}$ is assigned to sample document Y independently of samples for the other sentences $y^{(s')}$ the document likelihood can be written:

$$\frac{\partial}{\partial \theta_i} \log P(Y|X;\theta) = \frac{\partial}{\partial \theta_i} \sum_{y^{(s)} \in Y} \log P(y^{(s)}|x^{(s)};\theta) \tag{5.11}$$

Consequently $y_n^{(s)}$ – the n^{th} sample for the s^{th} sentence in each batch-level document – contributes the following term to the overall gradient:

$$\frac{\alpha}{N-1} \sum_{Y: y^{(s)} = y_n^{(s)}} (D(Y,Y^*) - \overline{D(Y,Y^*)}) \frac{\partial}{\partial \theta_i} \log P(y_n^{(s)}|x^{(s)};\theta) \tag{5.12}$$

In other words the gradient of each sample is weighted by the aggregated document-level scores for documents in which the sample appears. Compare with Eq. 5.8, assuming each sample appears in exactly one document. We have shown that under doc-MRT, each sentence-level sample has the same gradient contribution as for a direct implementation of sequence-level MRT with sequence-level metric Δ replaced by document-level metric D calculated over the appropriate samples. The additional design choices are in determining how to assign sampled sentences to sampled documents, and in obtaining the D metric over those documents.

Mini-batch document sampling

To generate sample documents we first sample sentences. Sentence sampling for NMT generates new word-level tokens in a left-to-right manner (Shen et al., 2016). In left-to-right generation each token is sampled from a distribution conditioned on previously sampled tokens, incidentally reducing the likelihood of exposure bias to gold references which the model cannot access at inference time (Ranzato et al., 2016). Sampling can be via beam search, or random sampling from the model distribution given previously sampled tokens. Beam search produces more likely samples which may be less diverse compared to random sampling (Edunov et al., 2018a).

Here we only consider sampling during training. While samples can be more easily generated offline with respect to fixed model parameters, such samples are not representative of the current model.

With N sample translations for each of the S sentence pairs per batch we can construct N^S possible sample documents as sequences of S sentences. Considering all possible documents is intractable unless N and S are small. It also carries the risk that a single sentence will appear in multiple sampled documents, giving it undue weight.

Instead we propose creating N documents by first ordering samples for each sentence (e.g. by sBLEU), then creating the n^{th} sample document Y_n by concatenating the n^{th} sample from each sentence. This gives a set of N diverse documents sampled from N^S possibilities. We expect the sampled documents to be diverse in contents, since a given sentence will only ever occur in a single document context, and diverse in score. We refer to this scheme as ordered document sampling.

Figure 5.2 illustrates document sampling by assigning sentences randomly to documents. Figure 5.3 shows a scheme where the same sentence samples are sorted for document-MRT (ordered).

5.3.2 Experimental setup

Data

We report first on English-German NMT. We initialize with a baseline trained on 17.5M sentence pairs from WMT19[2] News task datasets (Barrault et al., 2019), on which we learn a 32K-merge joint BPE vocabulary (Sennrich, Haddow, and Birch, 2016d). We validate on newstest2017, and evaluate on newstest2018.

[2]Note that this is a different training set than was used to train baselines for the EWC experiments in the previous section, resulting in different baseline BLEU scores on the same en-de test set.

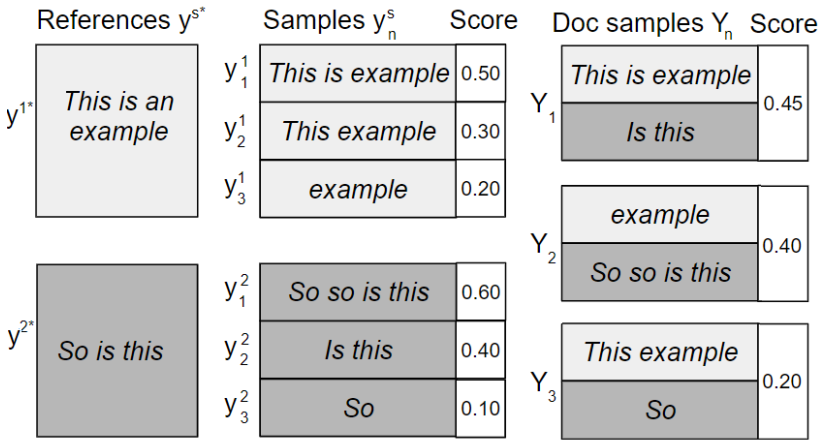


Fig. 5.2 **Seq-MRT** and **doc-MRT (random)** with $S = 2$ sentences / mini-batch and $N = 3$ samples / sentence, with illustrative (not real) scores. The original references are in the left column. In standard seq-MRT (middle) each sample has its own score (e.g. sBLEU). For doc-MRT (random) (right), samples are randomly assigned into N-wise 'documents', each with a combined score (e.g. document BLEU – in this example sequence scores are simply averaged). Document scores are on average less diverse with less distinct scores and a low likelihood of extreme distributions. However, they are less sensitive to individual samples, increasing robustness.

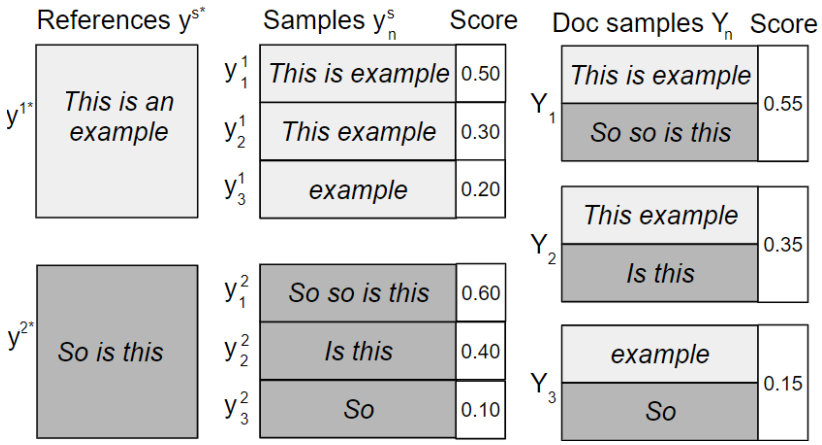


Fig. 5.3 The same example as Fig. 5.2, now comparing **seq-MRT** (middle) and **doc-MRT (ordered)** (right). For doc-MRT (ordered), we sort samples for a given sentence by quality (e.g. using sBLEU) before N-wise assignment into minibatch-level 'documents', each with a combined score. The doc-MRT scores are still less sensitive to individual samples, increasing robustness. However the ordered assignment enforces a more extreme range of combined costs, potentially a benefit to discriminative training.

We fine-tune on old WMT News task test sets (2008-2016) in two settings. With **random batches** sentences from different documents are shuffled randomly into mini-batches. In this case doc-MRT metrics are over pseudo-documents. With **document batches** each batch contains only sentences from one document, and doc-MRT uses true document context. We use the same experimental settings (e.g. batch size, sampling temperatures and Q function smoothing factors) for both forms of MRT for each experiment.

For Grammatical Error Correction (GEC) we train on sentences from NUCLE (Dahlmeier, Ng, and Wu, 2013) and Lang-8 Learner English (Mizumoto et al., 2012) with at least one correction, a total of 660K sentences. We evaluate on the JFLEG (Napoles, Sakaguchi, and Tetreault, 2017) and CoNLL 2014 (Ng et al., 2014) sets. For GEC experiments we use random batching only, since the data is not provided with document boundaries.

Finally we tune our systems for the WMT20 biomedical translation task with doc-MRT. Our experimental setup for this task is as described in Sec. 4.3.2.

Model, training and inference

For all models we use a Transformer model with the 'base' Tensor2Tensor parameters. We apply MRT only during fine-tuning, following previous work (Edunov et al., 2018a; Shen et al., 2016). In early experiments, we found that training from scratch with discriminative objectives (sequence- or document-based) is ineffective. We suspect samples produced early in training are so unlike the references that the model never receives a strong enough signal for effective training.

We train to validation set BLEU convergence on a single GPU. The batch size for baselines and MLE is 4K tokens. For MRT, where each sentence in the batch is sampled N times, we reduce batch size by N to keep computational requirements approximately the same. We then keep the effective batch size constant by delaying gradient updates by the same factor (Saunders, Stahlberg, de Gispert, et al., 2018).

We generate sequence samples for MRT by autoregressive sampling with temperature τ, where $\tau = 0$ would correspond to sampling the most probable target sentence under the model N times for each source sentence. We select both τ and α, the Q-function smoothing factor, by grid search over validation results for both seq- and doc-MRT, finding that $\tau = 0.3$ and $\alpha = 0.6$ give good performance for both schemes.

At inference time we decode using beam size 4 using SGNMT. News task BLEU scores are for cased, detokenized output, calculated using SacreBLEU. Biomedical test scores are case-insensitive BLEU for 'OK' sentences from the 2020 biomedical translation task test set as reported by the organizers (Bawden, Di Nunzio, et al., 2020).

Computation and sample count

Our proposed document-MRT approach is slightly more complex than sequence-MRT due to the additional score ordering and aggregation steps. In practice we find that this extra computation relating to the sequence-level scores is negligible when compared to the computational cost of sentence sampling, required for all forms of MRT.

Our MRT experiments use $N = 8$ random samples per sentence unless otherwise stated. In this we choose the highest N we can practically experiment with, since previous work finds MRT performance increasing steadily with more samples per sentence (Shen et al., 2016).

That we see improvements with so few samples is in contrast to previous work which finds BLEU gains only with 20 or more samples per sentence for sequence-MRT (Edunov et al., 2018a; Shen et al., 2016). However, we find that document-MRT allows improvements with far fewer samples, perhaps because the aggregation of scores over sentences in a context increases robustness to variation in individual samples.

Relatedly, we find that add-one BLEU smoothing (Lin and Och, 2004) is required for sequence-MRT as in Shen et al. (2016). However we find that doc-MRT can achieve good results without smoothing, perhaps because n-gram precisions are far less likely to be 0 when calculated over a document. This allows directly optimizing the BLEU metric rather than its approximation.

5.3.3 Document-level MRT experiments

MRT for NMT

In Table 5.6, we fine-tune an en-de baseline on documents from past News sets. We compare sentence-BLEU and document-BLEU MRT to fine-tuning with Maximum Likelihood Estimation (MLE).

MLE fine-tuning degrades the baseline. This suggests the baseline is well-converged, as is desirable for applying MRT (Shen et al., 2016). The degradation is slightly smaller with batches containing only sentences from the same document. We connect this to the idea that NMT batches with fewer sentence pairs have 'noisier' estimated gradients, harming training (Saunders, Stahlberg, de Gispert, et al., 2018). We expect batches of sentences from a single document to be similar and therefore give less noisy gradient estimates.

Both seq-MRT and doc-MRT improve over the baseline with $N = 8$. We also explore MRT at $N = 4$, with batch size adjusted as described in section 5.3.2 for the same effective batch size per update, and with fewer training steps such that the model 'sees' a similar proportion of the overall dataset. Early experiments selecting sentence samples via beam

Model	Random batches		Document batches	
Baseline	42.7			
MLE	40.0		41.0	
	$N=4$	$N=8$	$N=4$	$N=8$
Seq-MRT	42.6	43.5	42.6	43.5
Doc-MRT (random)	41.7*	43.1*	43.1	43.0
Doc-MRT (ordered)	**43.4**	**43.7**	**43.4**	**43.9**

Table 5.6 BLEU on en-de after MLE and MRT under $1-$sBLEU (seq-MRT) and $1-$ BLEU (doc-MRT). Results indicated by * are mean scores over 3 runs with the same settings, which had a range of just 0.2 BLEU.

.

Model	Random batches	Document batches
Baseline	39.2	39.2
MLE	41.2	40.0
Seq-MRT	39.4	40.5
Doc-MRT (ordered)	**39.0**	**38.9**

Table 5.7 TER on en-de after MLE and MRT under sentence-TER (seq-MRT) and doc-TER (doc-MRT). Lower TER is better.

search gave similarly poor results for both seq-MRT and doc-MRT. This may be because beam search produces insufficiently diverse samples for this task (Freitag and Al-Onaizan, 2017).

Sequence-MRT gives a 0.8 BLEU gain over the baseline with both batching schemes using $N = 8$ samples, but starts to degrade relative to the baseline with $N = 4$ samples. With document batches and $N = 8$, doc-MRT (ordered) outperforms seq-MRT by a further 0.4 BLEU. With $N = 4$ doc-MRT (ordered) still achieves a 0.7 BLEU improvement over the baseline, or a 0.8 BLEU improvement over seq-MRT. We suggest therefore that doc-MRT (ordered) may be a computationally more efficient alternative to seq-MRT when large sample counts are not practical.

For contrast with the ordered document sampling approach of Section 5.3.1, we give results for doc-MRT (random), which uses randomly sampled contexts. This approach falls significantly behind doc-MRT (ordered) with either batching scheme. Since doc-MRT (random) with random batches is exposed to randomness at the batch construction, sentence sampling and document sampling stages, these results are averages over 3 experimental runs, which gave fairly consistent results ($<$0.2 BLEU range across all runs). In general we do find that results with random batches and random ordering are variable and sensitive to batch size and batching scheme. However, we also conclude that the success of doc-MRT is not dependent on the presence of document boundaries in the training data.

Model	JFLEG				CONLL2014			
	P	R	M2	GLEU	P	R	M2	GLEU
Baseline	**67.3**	38.2	**58.4**	50.4	**54.4**	21.8	41.9	67.3
MLE	64.7	37.7	56.6	50.1	51.4	20.9	39.8	67.1
Seq-MRT	62.7	39.1	56.0	50.0	52.4	24.5	42.7	67.1
Doc-MRT (ordered)	64.4	**41.0**	57.8	**51.4**	53.2	**24.6**	**43.2**	**67.5**

Table 5.8 GEC Precision, Recall, M2, and GLEU after MLE and MRT. MRT is under $1-$sentence-GLEU for seq-MRT and $1-$doc-GLEU for doc-MRT. Both MRT schemes use random batches and random sentence sampling. Higher scores are better for all metrics.

We interpret these results by considering the effect on the per-sentence cost for the different schemes. We find MRT works well when sample scores are different enough to be discriminated, but suffers if scores are too different. This is in line with the findings of Edunov et al. (2018a) that including the gold reference causes the model to assign low relative probabilities to every other sample.

Doc-MRT aggregates scores over many samples, while seq-MRT uses individual scores. We believe this explains the stronger performance of doc-MRT for small values of N, especially for the ordered document scheme, which ensures scores are still different enough for MRT to discriminate.

Our approach can also be used with document-level metrics that are not intended to be used with individual sentences. In Table 5.7 we demonstrate this with TER, which estimates the edit rate required to correct a set of translation hypotheses. Document-TER MRT improves very slightly over a strong baseline, although batching scheme has less of an impact here. Notably seq-level MRT does not improve TER over the baseline, indicating TER may be too noisy a metric for use at the sentence level.

MRT for GEC

We next compare MRT approaches when tuning GEC systems under the GLEU metric (Napoles, Sakaguchi, Post, et al., 2015), an n-gram edit measure typically used at the corpus or document level. Table 5.8 shows that document MRT fine-tuning improves GLEU over the baseline, MLE fine-tuning, and a sequence-GLEU MRT formulation. Also notable is the change in M2, which finds the phrase-level edit sequence achieving the highest overlap with the gold-standard (Dahlmeier and Ng, 2012). MLE and sequence-MRT improve recall at a detriment to precision, suggesting over-generation of spurious corrections. Document-MRT likewise improves recall, but with a precision score closer to the baseline for more balanced performance. There is clear indication of a tension between M2 and GLEU: a small increase

in GLEU under doc-MRT on CONLL leads to a large increase in M2, while a large increase in GLEU under doc-MRT on JFLEG leads to a small decrease in M2.

We note that our improvements on JFLEG are similar to the improvements shown by Sakaguchi et al. (2017) for neural reinforcement learning with a sequence-GLEU cost metric. However, their results involve N=20 samples and 600k updates, compared to N=8 and 3k updates with our approach.

WMT20 biomedical task: addressing exposure bias with doc-MRT

In the previous chapter we discussed data-centric approaches to challenges in biomedical domain translation. In particular, we showed in Sec. 4.3 that fine-tuning generic pre-trained models on smaller amounts of biomedical-specific data can lead to strong performance very quickly. However, fine-tuning on small corpora exacerbates the effect of any noisy or poorly aligned sentence pairs. We treat this as a form of exposure bias, in that model overconfidence in training data results in poor translation hypotheses at test time, with concrete examples given in Table 4.7

We propose an approach to this problem in terms of the parameter fine-tuning scheme by using MRT. Wang and Sennrich (2020) have recently shown MRT as effective for combating exposure bias in the context of domain shift – test sentences which are very different from the training data. Our hypothesis in this section is that MRT is also more robust against over-exposure to misaligned training data.

We use memory-intensive large models for the biomedical task, and N is a limiting factor for MRT on such models. We have found that doc-MRT is particularly robust to small N, making this task an appropriate application for doc-MRT over seq-MRT.

The experimental setup, data preparation and MLE fine-tuning process are as described in Chapter 4. Here we discuss further experiments with MRT fine-tuning. We also discuss our WMT20 biomedical task test scores, since our submitted systems primarily involved MRT. For the test sentences, we additionally split any test lines containing multiple sentences before inference using the Python NLTK package[3], translate the split sentences separately, then remerged. We found this gave noticeable improvements in quality for the few sentences it applied to. While validation scores, as before, are for case-insensitive detokenized text obtained using SacreBLEU, test scores are as provided by the organizers for 'OK' sentences using Moses tokenization and the multi-eval tool.

We initialize both from baseline single models and from the baseline fine-tuned on the adaptation set with MLE. Experimentally, we find initializing fine-tuning from averaged

[3]https://pypi.org/project/nltk/ sentence splitter

		de2en	**en2de**	**es2en**	**en2es**
1	Baseline	38.8	30.6	48.5	46.6
2	MLE fine-tuning from 1	40.9	32.5	48.5	46.0
3	Fine-tuning from 1, no-title	40.9	32.2	47.0	44.9
4	Seq-MRT from 1	40.2	31.3	**49.0**	47.2
5	Seq-MRT from 2	41.2	32.7	45.9	43.9
6	Doc-MRT from 1	40.0	31.1	**49.0**	**47.4**
7	Doc-MRT from 2	41.3	**32.9**	45.8	43.3
8	Doc-MRT with no title from 3 (en-de) 1 (en-es)	**42.0**	32.4	48.5	46.9

Table 5.9 Validation BLEU developing models used in English-German and English-Spanish language pair submissions. Scores for single checkpoints. MRT fine-tuning from models 2 and 3 for Spanish-English did not improve over the baselines.

	de2en		**en2de**		**es2en**		**en2es**	
	Dev	**Test**	**Dev**	**Test**	**Dev**	**Test**	**Dev**	**Test**
MLE (all data) (en-de) / Baseline (en-es)	41.1	39.6	32.2	32.9	48.5	**46.6**	47.1	45.7
Doc-MRT (no-title data)	**41.9**	39.6	32.6	32.8	**49.0**	46.4	47.2	**46.7**
Doc-MRT (all data)	41.3	**39.8**	**33.0**	**33.2**	48.9	**46.6**	**47.7**	46.6

Table 5.10 Validation and test BLEU for models used in English-German and English-Spanish language pair submissions. All for averaged checkpoints. Test results are for 'OK' sentences as scored by the organizers.

checkpoints gives about the same or slightly worse results, with the added complexity of saving and averaging checkpoints.

Table 5.9 gives validation results for models fine-tuned with MRT, with lines 1-3 reproduced from Table 4.6. We confirm by comparing line 4 against 5 and line 6 against 7 that the convergence of the initializing model is very important for MRT. Initializing MRT fine-tuning with the English-German baseline results in a score 1.3 to 1.8 BLEU points lower than initializing with the MLE fine-tuned model. We see a similar pattern for English-Spanish, where the MLE fine-tuned model (line 2) underperforms the baseline (line 1). Consequently initialising MRT from the MLE tuned model (line 7) performs worse than initialising it from the baseline (line 6). Interestingly, seq-MRT performs nearly as well as doc-MRT here, perhaps because the models are larger and more strongly converged on in-domain data.

We submitted three runs to the WMT20 biomedical task for each language pair. All submissions used checkpoint averaging which we found generally improved or at least did not hurt performance. For en-de run 1 was the baseline model fine-tuned on MLE with all data, while for en-es we submitted the (also checkpoint averaged) baseline as MLE fine-tuning did not improve dev set performance. Run 2 was the run 1 model fine-tuned with doc-MRT on

System	German hypothesis	Approximate English translation
English source	[Associations of work-related strain with subjective sleep quality and individual daytime sleepiness].	
Human translation	[Zusammenhang von arbeitsbezogenen psychischen Beanspruchungsfolgen mit subjektiver Schlafqualität und individueller Tagesschläfrigkeit.]	
Baseline	[Assoziationen arbeitsbedingter Belastung mit subjektiver Schlafqualität und individueller Tagesschläfrigkeit].	[Associations of work-related stress with subjective sleep quality and individual daytime sleepiness.]
Fine-tuning	*Zusammenfassung.*	*Summary*
Doc-MRT	[Assoziationen arbeitsbedingter Belastung mit subjektiver Schlafqualität und individueller Tagesschläfrigkeit].	[Associations of work-related stress with subjective sleep quality and individual daytime sleepiness].
English source	[Effectiveness of upper body compression garments under competitive conditions: a randomized crossover study with elite canoeists with an additional case study].	
Human translation	[Effektivität von Oberkörperkompressionsbekleidung unter Wettkampfbedingungen: eine randomisierte Crossover-Studie an Elite-Kanusportlern mit einer zusätzlichen Einzelfallanalyse.]	
Baseline	[Wirksamkeit von Oberkörperkompressionsbekleidungsstücken unter kompetitiven Bedingungen: Eine randomisierte Crossover-Studie mit Elite-Kanuten mit einer *Additionsstudie*].	Effectiveness of upper body compression garments under competitive conditions: a randomized crossover study with elite canoeists with an *addition study*.
Fine-tuning	Eine randomisierte Crossover-Studie mit Elite-Kanuten mit einer *Additional Case Study wurde durchgeführt.*	A randomized crossover study with elite canoeists with an additional case study *was carried out*.
Doc-MRT	Eine randomisierte Crossover-Studie mit Elite-*Kanüsten* mit einer *Additional Case Study* hat zur Wirksamkeit von Oberkörperkompressionsbekleidung unter kompetitiven Bedingungen *geführt*.	A randomized crossover study with elite *canoes* with an additional case study *showed* the effectiveness of upper body compression garments under competitive conditions.

Table 5.11 Two sentences from the English-German 2020 test set with hypothesis translations from various models (title casing removed for clarity). Examples demonstrate the effects of exposure bias from fine-tuning on imperfectly aligned training sentences, compared to continued fine-tuning with MRT. Notable hypothesis departures from the reference are *emphasized*.

no-title data. Run 3 was the run 1 model fine-tuned with doc-MRT on all Medline abstract data. Table 5.10 gives scores for these submitted models.

Our best runs achieve the best and second-best BLEU-scored results among all systems for en2es and es2en respectively as reported by the organizers. For en-de our test scores are further behind other systems, perhaps indicating that the baseline system could have been stronger before fine-grained adaptation. This is also indicated by the strong improvement of these models under simple MLE.

We submitted the doc-MRT model on no-title data instead of the MLE on no-title data because MLE optimization did not improve over the baseline for en-es or en-es, with or without title lines, whereas MRT fine-tuning did. We also wanted to further examine whether doc-MRT was robust enough to benefit from 'noisy' data like the title lines, or whether cleaner no-title training data was more useful. In fact both forms of doc-MRT performed similarly on the test data, except in the case of en2de, where 'no-title' MRT scored 0.4 BLEU worse – further confirmation that source sentences with more information than the gold target can benefit MRT. We note that a MRT run was the best run or tied best run in all cases.

For the test runs, we additionally experimented with simply removing square bracket tokens from source sentences, since these could act as 'triggering' tokens for title sentences. This did seem to improve translations where applicable, as in the Table 4.7 examples. However, it is clearly not applicable to all forms of exposure bias, since it requires knowledge of all behaviours that could trigger exposure bias. MRT does not require such knowledge, but still reduces the effects of exposure bias (Table 5.11).

5.3.4 Document-level MRT summary

We present a novel approach for structured loss training with document-level objective functions. Our approach relies on a procedure for sampling a set of diverse batch-level contexts using N-wise sample ordering. As well as randomly selecting training data, we assess training with mini-batches consisting only of single document contexts. While the scope of this work does not extend to sampling sentences given document context, this would be an interesting direction for future work.

We demonstrate improvements covering three document-level evaluation metrics: BLEU and TER for NMT and GLEU for GEC. We also find that Minimum Risk Training can benefit from imperfectly aligned training examples while reducing the effects of exposure bias. We finish by noting that the original MERT procedure developed for SMT optimized document-level BLEU and with our procedure we reintroduce this to NMT.

5.4 Conclusions

This chapter presents approaches to changing NMT model behaviour by varying the adaptation procedure. The data-centric approaches to NMT domain adaptation discussed in Chapter 4 can lead to performance degradation on previously learned domains due to forgetting, or on a new domain due to exposure bias.

We address the forgetting problem by applying EWC regularization during NMT adaptation. We do so while adapting to a single new domain, and while adapting to two new domains sequentially. We show that EWC reduces forgetting. As well, depending on the relationship between pre-training and new domains, EWC can lead to stronger performance on either new or original domain than tuning without regularization.

Separately, we develop a fine-tuning scheme based on MRT that takes into account the use of corpus-level metrics in NMT, as well as other tasks. We find that our doc-MRT scheme gives stronger results than standard seq-MRT with small sample count, making it more efficient to use for large models. We also find that our scheme mitigates previously described problems of exposure bias in the biomedical translation task.

Regularized adaptation allows strong performance by a single translation model over multiple domains of interest. However, we note that some trade-off is still likely between improved performance on the new domain and forgetting on previously-learned domains. In the next chapter, we will explore effective use of multiple single-domain or multi-domain models to translate multiple domains of interest by adjusting the inference procedure.

Chapter 6

Inference schemes to combine benefits of adapted NMT models

This chapter draws from the following publications: Saunders, Stahlberg, de Gispert, et al. (2019) throughout and Saunders, Stahlberg, and Byrne (2019) in Sec. 6.3

6.1 Motivation

The previous chapters have shown that it is possible, with careful data selection or adaptation schemes, to produce strong domain-specific translation models. These models may be adapted to extremely narrow domains, for example by fine-tuning on a small or otherwise easily-fitted dataset. Narrow domain models, as in the biomedical task experiments, may however give poor translations for sentences from other domains. Alternatively models may be adapted to translate multiple domains, for example using regularized adaptation techniques. Models adapted to give good performance over multiple domains, as in regularization experiments, usually show some level of trade-off between new-domain and pre-training domain performance.

If the goal is optimal translation in all scenarios, it may be beneficial to use ensembles of separate models at inference time (Sec. 2.4.3). In this chapter we explore inference schemes to combine benefits of adapted domain-specific NMT models.

Translating with an ensemble of models is slower and involves more memory to store or run than using a single NMT model. However, ensembles of NMT models typically perform better: the best achieving systems in translation evaluation campaigns are consistently ensembles (Barrault et al., 2019). Moreover an ensemble of models trained on multiple domains may achieve good performance over all domains, as we showed in Sec. 4.2.

As reviewed in Sec. 3.5.1, model ensembling has previously been applied to domain adaptation scenarios. A typical example is the approach of Freitag and Al-Onaizan (2016), who use an ensemble of an in-domain and out-of-domain model to translate all sentences. However, prior work on this problem makes the implicit assumption that a good ensemble weighting can be pre-determined for all test sentences, before any test sentences are available. We aim to address that assumption in this chapter, as well as our third research question on domain-specific NMT in scenarios where the exact domain of a test sentence is not pre-determined.

In Sec. 6.2 we compare typical 'static' approaches to multi-domain ensemble weighting with our own language modelling approach to conditioning ensemble weights on the source sentence, allowing per-sentence ensemble weighting. In Sec. 6.3 we develop adaptive ensemble weighting for NMT, in which the contribution of different ensemble models can vary while translating a single sentence as well as between sentences.

6.2 Language-model interpolated ensembles

In this section we focus on the potential benefits of performing inference with a weighted ensemble of models trained on different domains. An appropriate ensemble weighting will depend on the test domain. In reality any test sentence may be drawn from some unknown domain, in which case a good weighting may not be obvious. Crucially, we assume the realistic scenario where the domain is unknown at inference time. In particular, we address the assumption that a fixed ensemble weighting should be determined according to the broad labels applied externally to a test set.

A related approach is described by Sajjad et al. (2017), who perform translation with a weighted multi-domain ensemble, where weights 'can be pre-defined or learned on a development set'. However, we make an important distinction between the domain of the training data and model – typically known, often with an available development set – and the test data, for which we cannot assume we know the domain or have access to some convenient set of relevant data.

6.2.1 Static decoder configurations for ensemble weighting

We assume we have models $1, ..., K$ such that each perform well on at least one domain, and that we have both training and development datasets for those K domains/models. We wish to weight each model's predictions by some static value W_k for each test sentence at inference time.

There are a number of existing approaches to this problem. One is the 'oracle' scenario, in which 1) the provenance of the test sentence is known and 2) the appropriate model to translate a sentence of that provenance is known. This the approach taken when, for example, we decide translate the IT test set with the best IT model in Sec. 5.2. This may be practical in limited scenarios such as translation shared tasks where the training and test domains are provided, but is not applicable in general.

Other approaches include simply taking a uniform ensemble of all models regardless of sentence (Freitag and Al-Onaizan, 2016) or tuning ensemble weights on a development set (Sajjad et al., 2017). The former approach is simple, but does not guarantee good performance on any particular domain. The latter approach can be targeted to a particular domain, but weight tuning can be very slow, and the approach requires either that the domain of the test set is known or that a development set representative of the test set is available.

Instead we propose a simple source-sentence conditioning approach, which we refer to as ensembling with an Informative Source (IS). We train language models on source training sentences from each domain k. We then obtain weights W_k which are static during inference, but which are defined separately for each test source sentence $\mathbf{x}$:

$$W_k(\mathbf{x}) = \frac{P_{LM_k}(\mathbf{x})}{\sum_{k'} P_{LM_{k'}}(\mathbf{x})} \tag{6.1}$$

6.2.2 Experimental setup

Throughout this chapter we report on Spanish-to-English (es-en) and English-to-German (en-de) translation, using the same experimental setup, data and models as described in Sec. 5.2.2. The language models W_k used for IS scoring are 4-gram language models trained on the source training data for each model domain k. Each language model is estimated with modified Kneser-Ney smoothing (Heafield et al., 2013) and without n-gram pruning using the KenLM toolkit (Heafield, 2011).

6.2.3 Informative source ensemble weighting experiments

Per-sentence source conditioning is more effective than tuning for less computation

We first demonstrate the power of our IS approach by comparing to weight tuning on validation sets for the Scielo Health/Bio domain task in Table 6.1. The results demonstrate that the optimal ensemble weighting for a set of domains is not necessarily intuitive. For the Health domain, the oracle choice – translating all Health test sentences with the corresponding model only – gives the best result. However, for the Bio domain the oracle choice performs

comparatively poorly. Instead, placing about a third of the ensemble weighting onto the Health model results in an additional 1 BLEU on the Bio dev set. IS weighting matches or exceeds the best tuned score in each case.

With the combined BLEU column we simulate the realistic scenario where domain labels for some mixed-domain test set are not available. In this case, uniform ensembling is a reasonable approach, outperforming either oracle model overall. However, IS weighting still outperforms the best tuned ensemble by 0.8 BLEU.

IS often selects one 'correct' model, but not necessarily the oracle

Analysis of the scores suggests that many of Health and Bio domain validation sentences *are* best translated by only one of the models – about 80% of the weights determined by IS are 0.99 or higher for one domain. This information allows a potentially significant inference speed improvement, since these sentences can simply be translated by a single model instead of by ensemble decoding with no performance loss. As well, we note that extremely high weights do not necessarily correspond to the domain label. Of the Health set 3% of sentences will be translated by the Bio model alone under IS, and 38% of the Bio validation set will be translated by the Health model alone under IS. The strong results under IS therefore support our hypothesis that provenance should not be relied upon as a surrogate for test sentence domain.

$\mathbf{W}_{\text{Health}}$ $\mathbf{W}_{\text{Bio}}$	**Health dev BLEU**	**Bio dev BLEU**	**Combined dev BLEU**
IS	**37.2**	**40.0**	**38.7**
0.0 1.0 (Bio model only)	31.7	38.7	35.5
0.1 0.9	32.4	39.3	36.2
0.2 0.8	33.2	39.7	36.7
0.3 0.7	33.8	39.7	37.0
0.4 0.6	34.5	39.7	37.4
0.5 0.5 (Uniform ensemble)	35.2	39.6	37.8
0.6 0.4	35.9	39.5	37.9
0.7 0.3	36.3	39.1	37.9
0.8 0.2	36.7	38.8	37.9
0.9 0.1	37.1	37.9	37.6
1.0 0 (Health model only)	**37.2**	37.7	37.5

Table 6.1 Validation BLEU for statically interpolated ensembles between the Scielo es-en Health and Bio models, compared to per-sentence IS weighting.

We note that a significant proportion of the weights do involve meaningful contributions from more than one ensemble model. Table 6.2 gives sentences and weights for sentences from a biomedical dev set for English-to-German translation models trained on News and

Biomedical data, two more distant domains. The non-oracle weightings in these cases have intuitive interpretations: all three sentences are from the biomedical domain, but the first could also be interpreted as a sentence from a dialog or Q&A domain, the second from a software or IT domain, the third from an interview or diary domain.

We also emphasize that the validation set cannot always be taken as representative of the test set. The IS weightings for the Health test set mark 79% of sentences for oracle translation and 5% for Bio-only translation, quite similar to the dev set proportions of 82:3 for oracle:other. However, 58% of the Bio test set is marked for oracle translation and 14% for Health-only translation, a significant shift from the 43:38 dev set proportions.

Source sentence	$\mathbf{W_{Bio}}$	$\mathbf{W_{News}}$
Where can I find information about diagnosis or management of Gaucher's disease?	0.64	0.36
The data will then be entered into a database and analyzed.	0.56	0.44
I have to check my blood sugars more frequently when I'm playing sport.	0.14	0.86

Table 6.2 Ensemble model weights under the IS scheme for the English-to-German Biomedical and News NMT models. All source sentences are from Khresmoi medical article summary set (Dušek et al., 2017)

Decoder scheme	**es-en**		**en-de**		
	Health	**Bio**	**News**	**TED**	**IT**
Oracle model	35.9	36.1	**37.8**	24.1	39.6
Uniform	33.1	36.4	21.9	18.4	38.9
IS	**36.0**	**36.8**	37.5	**25.6**	**43.3**

Table 6.3 Test BLEU for 2-model es-en and 3-model en-de ensembles of single-domain (unadapted) models from Sec. 5.2.2, compared to results with the oracle model chosen to correspond to the test domain. Uniform ensembling generally underperforms the oracle, while IS can significantly outperform the oracle.

Table 6.3 contains test BLEU for two- and three-model ensembling (models 1+2 from Table 5.2 and 1+2+3 from Table 5.3). We find that IS significantly outperforms uniform ensembling. IS also outperforms the oracle in all cases except en-de News, indicating this scheme does not simply select a single model for each test set.

6.2.4 Ensembling with static interpolation: summary

In this section we report on weighting mixed-domain ensembles for multi-domain NMT inference. We determine static ensemble weights for each test sentence at inference time using n-gram language model scores. Our approach significantly out-performs uniform ensembling,

and often outperforms the 'oracle' model trained on data with the same provenance as the test sentence. Moreover, our approach does not require knowledge of test set provenance or validation data availability.

6.3 Bayesian Interpolation for adaptive ensembles

In this section we continue the chapter's theme of adaptive inference by exploring non-static weighting for multi-domain ensembles. The previous section handled cases where the domain of a given test sentence is unknown, even if the test set has a domain label. In this section we further allow the domain weighting for a given test sentence to vary within that sentence.

We develop an adaptive inference scheme for NMT ensembles by extending Bayesian Interpolation (BI) (Allauzen and Riley (2011), reviewed in Sec. 3.5.1) to sequence-to-sequence models.[1] This lets us calculate ensemble weights adaptively over time without needing the domain label, giving strong improvements over uniform ensembling for baseline and fine-tuned models. We show that our approach is complementary to the static decoder configurations described in the previous section, as well as the regularized training techniques described in the previous chapter.

6.3.1 Adaptive decoding

We extend the BI formalism to condition on a source sequence, letting us apply it to adaptive NMT ensemble weighting. The derivation below broadly follows that in Sec. 3.5.1, but we introduce dependence on source sentence $\mathbf{x}$ and discuss the resulting modelling choices.

We consider models $p_k(\mathbf{y}|\mathbf{x})$ trained on K distinct domains, used for tasks $t = 1,\ldots,T$. In this case we say that a task is decoding from one domain, so $T = K$. We assume throughout that $p(t) = \frac{1}{T}$, i.e. that tasks are equally likely absent any other information. A standard, fixed-weight ensemble would translate with:

$$\underset{\mathbf{y}}{\operatorname{argmax}}\, p(\mathbf{y}|\mathbf{x}) = \underset{\mathbf{y}}{\operatorname{argmax}} \sum_{k=1}^{K} W_k p_k(\mathbf{y}|\mathbf{x}) \tag{6.2}$$

The BI formalism assumes that we have tuned sets of ensemble weights $\lambda_{k,t}$ for each task. We can define a task-conditional ensemble:

$$p(\mathbf{y}|\mathbf{x},t) = \sum_{k=1}^{K} \lambda_{k,t}\ p_k(\mathbf{y}|\mathbf{x}) \tag{6.3}$$

[1] See `bayesian` combination schemes at https://github.com/ucam-smt/sgnmt

which can be used as a fixed weight ensemble if the task is known. However if the task t is not known, we wish to translate with:

$$\underset{\mathbf{y}}{\operatorname{argmax}}\, p(\mathbf{y}|\mathbf{x}) = \underset{\mathbf{y}}{\operatorname{argmax}} \sum_{t=1}^{T} p(t,\mathbf{y}|\mathbf{x}) \tag{6.4}$$

At decoder step i, where h_i is the history $\mathbf{y}_{1:i-1}$, we generate the i^{th} token in the translation hypothesis:

$$\begin{aligned} p(y_i|h_i,\mathbf{x}) &= \sum_{t=1}^{T} p(t,y_i|h_i,\mathbf{x}) \\ &= \sum_{t=1}^{T} p(t|h_i,\mathbf{x})\, p(y_i|h_i,t,\mathbf{x}) \end{aligned} \tag{6.5}$$

We now introduce the expression for the task-conditional output (Eq 6.3) and marginalize over tasks t to find adaptive weights $W_{k,i}$:

$$\begin{aligned} p(y_i|h_i,\mathbf{x}) &= \sum_{t=1}^{T} p(t|h_i,\mathbf{x}) \sum_{k=1}^{K} \lambda_{k,t}\, p_k(y_i|h_i,\mathbf{x}) \\ &= \sum_{k=1}^{K} p_k(y_i|h_i,\mathbf{x}) \sum_{t=1}^{T} p(t|h_i,\mathbf{x})\lambda_{k,t} \\ &= \sum_{k=1}^{K} W_{k,i}\, p_k(y_i|h_i,\mathbf{x}) \end{aligned} \tag{6.6}$$

The final line of Eq. 6.6, by comparison with Eq. 6.2, has the form of an adaptively weighted ensemble where:

$$W_{k,i} = \sum_{t=1}^{T} p(t|h_i,\mathbf{x})\lambda_{k,t} \tag{6.7}$$

In decoding, ensemble weight adaptation at each step i relies on a recomputed estimate of the *task posterior*, where $p(h_i|t,\mathbf{x})$ is simply equivalent to the task-conditional probability of the previous output token $p(y_{i-1}|t,\mathbf{x})$:

$$p(t|h_i,\mathbf{x}) = \frac{p(h_i|t,\mathbf{x})p(t|\mathbf{x})}{\sum_{t'=1}^{T} p(h_i|t',\mathbf{x})p(t'|\mathbf{x})} \tag{6.8}$$

Static decoder configurations

In static decoding (Eq. 6.2), the weights W_k are constant for each source sentence $\mathbf{x}$. These static weights can be obtained under the BI formalism given certain assumptions. The BI

ensemble weights are determined by $p(t|\mathbf{x})$, the probability of the task conditioned only on the source sentence, and by ensemble weights $\lambda_{k,t}$, which determine the contribution from domain k model to inference for task t.

Consider $\lambda_{k,t} = p(t|\mathbf{x}) = \frac{1}{T}$: the task distribution is not affected by the source sentence, and all model domains contribute equally to all possible tasks. From Eq. 6.7 the ensemble weight for model k at decoder step i becomes $W_{k,i} = \sum_{t=1}^{T} \frac{1}{T}\frac{1}{T} = \frac{1}{T} = \frac{1}{K}$ where $T = K$. In other words, these assumptions lead to a fixed equal-weight interpolation of the component models: uniform ensembling.

Less general assumptions lead to static decoding with task posteriors conditioned only on the source sentence, as we discussed in the previous section. In the context of BI this reflects an assumption that the inference history can be disregarded and that $p(t|h_i, \mathbf{x}) = p(t|\mathbf{x})$. In the most straightforward case, we assume that only domain k is useful for task t: $\lambda_{k,t} = \delta_k(t)$ (1 for $k = t$, 0 otherwise). This simplifies to a fixed ensemble:

$$W_k = p(k|\mathbf{x}) \tag{6.9}$$

and decoding proceeds according to Eq. 6.7. This is the informative source (IS) scenario described in the previous section. As before we use source language n-gram language models to estimate $p(t = k|\mathbf{x})$ in Eq. 6.9.

Adaptive decoder configurations

For adaptive decoding with Bayesian Interpolation, as in Eq. 6.6, the model weights vary during decoding according to Eq. 6.7 and Eq. 6.8, assuming that $p(t|\mathbf{x}) = p(t) = \frac{1}{T}$. This corresponds to the approach in Allauzen and Riley (2011), which considers only language model combination for speech recognition. We refer to this in experiments simply as BI.

A refinement if $T = K$ is to incorporate Eq. 6.1 into Eq. 6.8, for $p(t|\mathbf{x}) = p(t = k|\mathbf{x}) = W_k$ as defined under the IS formulation. We refer to this as Bayesian Interpolation with an informative source (BI+IS).

We now address the choice of $\lambda_{k,t}$. A simple but restrictive approach is to take $\lambda_{k,t} = \delta_k(t)$. We refer to this as *identity-BI*, and it embodies the assumption that only one domain is useful for each task. Unlike IS, which has the same λ, updating $p(t|h_i, \mathbf{x})$ means that weights are still adaptive during decoding. We also note that unlike the oracle inference case, we do not need to specify the task/domain corresponding to each test sentence.

Alternatively, if we have validation data V_t for each task t, parameter search can be done to optimize $\lambda_{k,t}$ for BLEU over V_t for each task. This is straightforward but relatively costly.

	Decoder	$\mathbf{p(t|x)}$	$\lambda_{\mathbf{k,t}}$
Static	Uniform	$\frac{1}{T}$	$\frac{1}{T}$
	IS	$\frac{P_{LM_k}(x)}{\sum_{k'} P_{LM_{k'}}(x)}$	$\delta_k(t)$
Adaptive	Identity-BI	$\frac{1}{T}$	$\delta_k(t)$
	BI	$\frac{1}{T}$	$\frac{\overline{P_{LM_{k,t}}}}{\sum_{k'} \overline{P_{LM_{k',t}}}}$
	BI+IS	$\frac{P_{LM_k}(x)}{\sum_{k'} P_{LM_{k'}}(x)}$	$\frac{\overline{P_{LM_{k,t}}}}{\sum_{k'} \overline{P_{LM_{k',t}}}}$

Table 6.4 Setting task posterior $p(t|\mathbf{x})$ and domain-task weight $\lambda_{k,t}$ for T tasks under decoding schemes in this work. Note that IS can be combined with either Identity-BI or BI by simply adjusting $p(t|h_i,\mathbf{x})$ according to Eq. 6.8. $\overline{P_{LM_{k,t}}}$ is as defined in Eq. 6.10.

We propose a simpler approach based on the source language n-gram language models from Eq. 6.1. We assume that each G_t is also a language model for its corresponding domain k. With $\overline{P_{LM_{k,t}}} = \sum_{\mathbf{x} \in V_t} P_{LM_k}(\mathbf{x})$, we take:

$$\lambda_{k,t} = \frac{\overline{P_{LM_{k,t}}}}{\sum_{k'} \overline{P_{LM_{k',t}}}} \tag{6.10}$$

$\lambda_{k,t}$ can be interpreted as the probability that task t contains source sentences $\mathbf{x}$ drawn from domain k as estimated over the set of sentences V_t. We note that unlike optimizing $\lambda_{k,t}$ for BLEU over a validation set, this approach does not require access to reference sentences. V_t could in principle be the test set or test document rather than some additional validation set.

Figure 6.1 demonstrates this scheme when weighting a biomedical and a general (news) domain model to produce biomedical sentences under BI. In the first example the model weights $W_{k,i}$ are even until biomedical-specific vocabulary is produced, at which point the in-domain model dominates. The other examples contain less 'biomedical-specific' terminology and so the adaptive weights stay approximately even during decoding. We note that the adaptive weights reflect the overall language model weights found using IS in Table 6.2.

Summary

We summarize our approaches to decoding in Table 6.4.

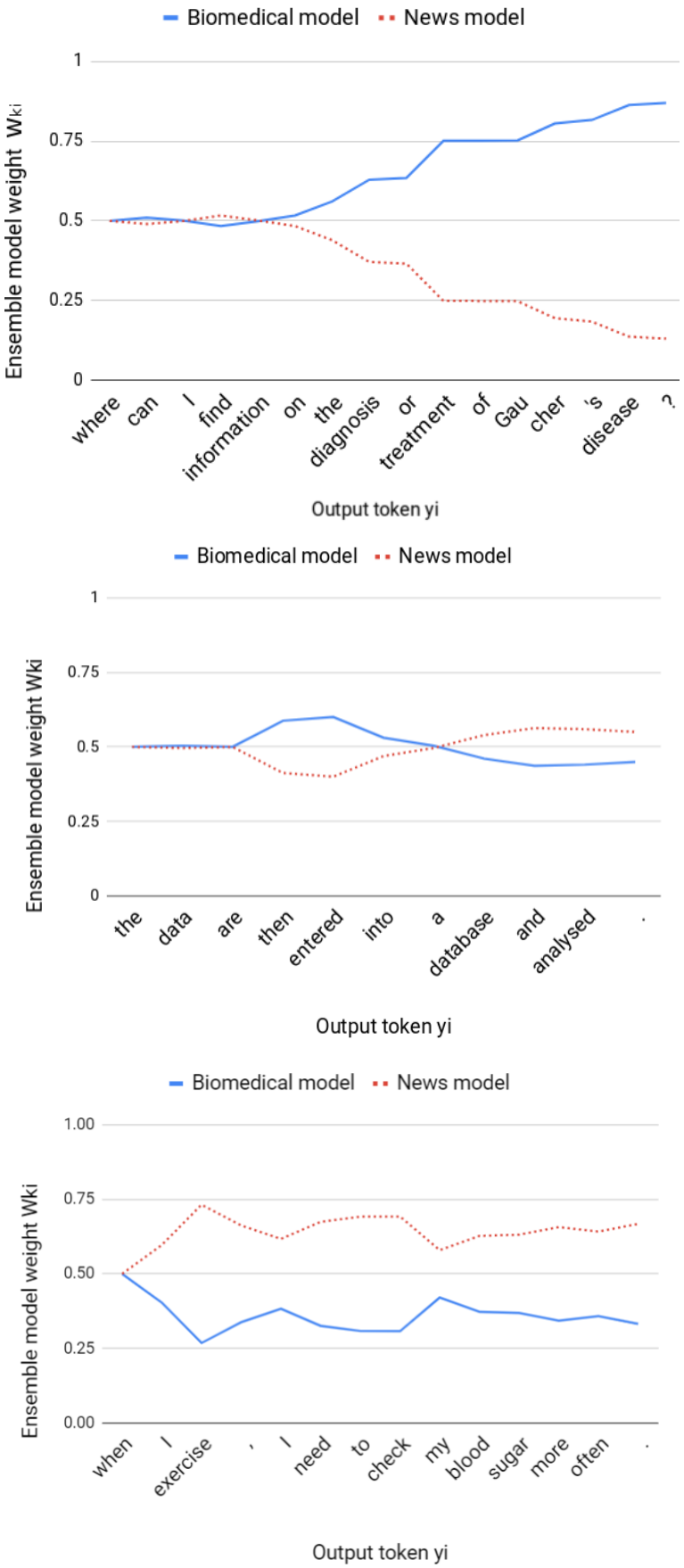


Fig. 6.1 Adaptively adjusting ensemble model weights $W_{k,i}$ (Eq. 6.7) during decoding with Bayesian Interpolation for German-to-English Khresmoi sentences.

Experimental setup

We primarily use the same experimental setup as in the previous section. Models that have been tuned with domain adaptation techniques are from Sec. 5.2. For the experiments relating to the biomedical translation task we use the same experimental setup, data and models as in Sec. 4.2.2.

As in the previous section we use language models trained on the source training data for each domain using the KenLM toolkit (Heafield, 2011). We use 4-gram models for all cases except for the biomedical task, where we use 2-gram models. This is because there was a high likelihood of the test data having little topic overlap with any training data, so we wished to avoid unduly weighting sparse higher-order n-grams across the similar domains.

6.3.2 Adaptive decoding experiments

History conditioning with BI is complementary to source conditioning with IS

Table 6.5 gives results using our adaptive decoding schemes with ensembles of unadapted models trained only on one domain, as in the previous section. We again compare with the 'oracle' model trained on each domain, which we can only use if we know the test domain. We also compare to results with uniform ensembling and IS alone.

Identity-BI strongly improves over uniform ensembling, especially on the general domains. BI with λ as in Eq. 6.10 improves further for all test sets except es-en Bio. BI individually matches or outperforms the oracle in all cases, indicating that, like IS, the BI scheme does not simply select a single model. BI generally performs about the same or slightly better than IS.

The combined scheme of BI+IS outperforms either BI or IS individually, except for en-de IT. We speculate IT is a distinct enough domain that $p(t|x)$ has little effect on adapted BI weights.

Bayian Interpolation improves over static ensembles of EWC-adapted models

In Table 6.6 we apply the best adaptive decoding scheme, BI+IS, to models fine-tuned with EWC. The es-en ensemble consists of models 1+6 from Table 5.2 and the en-de ensemble models 1+7+10 from Table 5.3. As described in Section 5.2.2, EWC models perform well over multiple domains, so the improvement over uniform ensembling is less striking than for unadapted models. Nevertheless adaptive decoding improves over both uniform ensembling and the oracle model in most cases.

Decoder configuration	es-en		en-de		
	Health	Bio	News	TED	IT
Oracle model	35.9	36.1	37.8	24.1	39.6
Uniform	33.1	36.4	21.9	18.4	38.9
Identity-BI	35.0	36.6	32.7	25.3	42.6
BI	35.9	36.5	38.0	26.1	**44.7**
IS	**36.0**	36.8	37.5	25.6	43.3
BI + IS	**36.0**	**36.9**	**38.4**	**26.4**	**44.7**

Table 6.5 Test BLEU for 2-component es-en ensembles and 3-component en-de ensembles, compared to oracle model chosen if test domain is known. All models are trained on a single domain, without fine-tuning. BI and IS are complementary ensemble weighting schemes.

Decoder configuration	es-en		en-de		
	Health	Bio	News	TED	IT
Oracle model	35.9	37.8	37.8	27.0	57.0
Uniform	36.0	36.4	**38.9**	26.0	43.5
BI + IS	**36.2**	**38.0**	38.7	**26.1**	**56.4**

Table 6.6 Test BLEU for 2-model es-en and 3-model en-de model ensembling for models adapted with EWC, compared to oracle model last trained on each domain, chosen if test domain is known. Best results without oracle information in bold. BI+IS outperforms uniform ensembling and in some cases outperforms the oracle.

			Decoder configuration	
Language pair	Model type	Oracle model	Uniform	BI + IS
es-en	Unadapted	36.4	34.7	36.6
	No-reg	36.6	34.8	-
	EWC	37.0	36.3	**37.2**
en-de	Unadapted	36.4	26.8	38.8
	No-reg	41.7	31.8	-
	EWC	42.1	38.6	**42.0**

Table 6.7 Total BLEU for test data concatenated across domains. Results from 2-model es-en and 3-model en-de ensembles, compared to oracle model chosen if test domain is known. Best results without oracle information in bold. No-reg uniform corresponds to the approach of Freitag and Al-Onaizan (2016). BI+IS performs similarly to strong oracles with no test domain labeling.

With adaptive decoding, we do not need to assume whether a uniform ensemble or a single model might perform better for some potentially unknown domain. We highlight this in Table 6.7 by reporting results with the ensembles of Tables 6.5 and 6.6 over concatenated test sets, to mimic the realistic scenario of unlabelled test data. We additionally include the

uniform no-reg ensembling approach given in Freitag and Al-Onaizan (2016) using models 1+4 from Table 5.2 and 1+5+8 from Table 5.3.

Uniform no-reg ensembling outperforms unadapted uniform ensembling, since fine-tuning gives better in-domain performance. EWC achieves similar or better in-domain results to no-reg while reducing forgetting, resulting in better uniform ensemble performance than no-reg.

BI+IS decoding with single-domain trained models achieves gains over both the simple uniform approach and over oracle single-domain models. BI+IS with EWC-adapted models gives a 0.9 / 3.4 BLEU gain over the strong uniform EWC ensemble, and a 2.4 / 10.2 overall BLEU gain over the approach described in Freitag and Al-Onaizan (2016).

Adaptive inference at the WMT19 Biomedical translation task

As described in Chapter 4 we initially approached the 2019 WMT Biomedical translation task using transfer learning to obtain a series of strong NMT models on distinct domains. We then combined those models into multi-domain ensembles. Here we further experiment with an adaptive language-model ensemble weighting scheme. Our final submission achieved the best submitted test BLEU scores on both directions of English-Spanish translation.

The domain of individual documents in the 2019 Medline test dataset is unknown, and may vary sentence-to-sentence. While we have seen that uniformly-weighted ensembles of models from different domains can give good results in this case, we suggest a better approach would take into account the likely domain, or domains, of each test sentence. We therefore investigate applying Bayesian Interpolation for language-model based multi-domain ensemble weighting.

In this case we use a power smoothing scheme, since the tuned model domains are very similar and we do not wish to over-weight a particular domain.

$$W_k(x) = \frac{P_{LM_k}(x)^\alpha}{\sum_{k'} P_{LM_{k'}}(x)^\alpha} \tag{6.11}$$

Here α is a smoothing parameter. Uniform ensembling corresponds to $\alpha = 0.0$ and unsmoothed IS corresponds to $\alpha = 1.0$.

For validation results we report cased BLEU scores with SacreBLEU (Post, 2018); test results use case-insensitive BLEU.

We submitted three runs to the WMT19 biomedical task for each language pair: the best single all-biomed model, a uniform ensemble of models on two en-de and three es-en domains, and an ensemble with Bayesian Interpolation. Tables 6.8 and 6.9 give validation and test scores.

	es2en				en2es			
	Khresmoi	**Health**	**Bio**	**Test**	**Khresmoi**	**Health**	**Bio**	**Test**
1: Health → All-biomed	52.1	36.7	37.0	42.4	44.2	35.0	39.0	44.9
1 → Health	51.1	**37.0**	37.2	-	44.0	**36.3**	39.5	-
1 → Bio	50.6	36.0	38.0	-	**45.2**	35.3	**41.3**	-
Uniform ensemble	**52.2**	36.9	37.9	**43.0**	45.1	35.6	40.2	45.4
BI ensemble (α=0.5)	52.1	**37.0**	**38.1**	42.9	44.5	35.7	41.2	**45.6**

Table 6.8 Validation and test BLEU for models involved in English-Spanish language pair submissions.

	de2en			en2de		
	Khresmoi	**Cochrane**	**Test**	**Khresmoi**	**Cochrane**	**Test**
News	43.8	46.8	-	30.4	40.7	-
News → All-biomed	44.5	47.6	27.4	31.1	39.5	26.5
Uniform ensemble	45.3	48.4	**28.6**	**32.6**	42.9	**27.2**
BI ensemble (α=0.5)	**45.4**	**48.8**	28.5	32.4	**43.1**	26.4

Table 6.9 Validation and test BLEU for models used in English-German language pair submissions.

We find that a uniform multi-domain ensemble performs well, giving 0.5-1.2 BLEU improvement on the test set over strong single models. We see small gains from using BI with ensembles on most validation sets, but only on en2es test.

Following test result release, we noted that, in general, we could predict BI ($\alpha = 0.5$) performance by comparing the uniform ensemble with the oracle model performing best on each validation domain. For en2es uniform ensembling underperforms the Health and Bio oracle models on their validation sets, and the uniform ensemble slightly underperforms BI on the test data. For en2de, by contrast, uniform ensembling is consistently better than oracles on the dev sets, and outperforms BI on the test data. For de2en and es2en, uniform ensembling performs similarly to the oracles, and performs similarly to BI.

	es2en	**en2es**	**de2en**	**en2de**
Uniform	**43.2**	45.3	28.3	25.9
BI (α=0.5)	43.0	**45.5**	28.2	25.2
BI (α=0.1)	**43.2**	**45.5**	**28.5**	**26.0**

Table 6.10 Comparing uniform ensembles and BI with varying smoothing factor on the WMT19 test data. Small deviations from official test scores on submitted runs are due to tokenization differences. $\alpha = 0.5$ was chosen for submission based on results on available development data.

From this, we hypothesize that BI ($\alpha = 0.5$) has a tendency to converge to a single model. This is effective when single models perform well (en2es) but ineffective if the uniform ensemble is predictably better than any single model (en2de). Consequently in Table 6.10 we experiment with BI ($\alpha = 0.1$). In this case BI matches or out-performs the uniform ensemble. Notably, for en2es, where BI ($\alpha = 0.5$) performed well, taking $\alpha = 0.1$ does not harm performance.

6.3.3 Adaptive ensembling summary

In this section we report on domain adaptive inference where sentence weights can vary during inference on a single sentence. We compare various schemes to determine adaptive inference hyperparameters, and find that a language model weighting out-performs an 'identity' assumption that a single model performs best on a given domain. Once again, our adaptive inference approaches significantly out-perform uniform ensembling and 'oracle' single-domain models. Our approach also matches oracle performance when ensembling models tuned for multi-domain performance using EWC.

6.4 Conclusions

This chapter presents our adaptive inference procedure for multi-domain ensembles. Data-centric and adaptation-centric approaches discussed in Chapters 4 and 5 either assume foreknowledge of the test domain, or involve some trade-off in performance over multiple possible test domains. Instead we report on decoding techniques that adapt NMT to new domains while preserving performance on the original domain.

We first explore static interpolation with simple language model weighting conditioned on source test sentences. This involves less computation than tuning ensemble weights on validation set via grid search, and is more effective. It also does not assume the availability of a validation set.

We then extend Bayesian Interpolation with source information and apply it to NMT decoding with unadapted and fine-tuned models, adaptively weighting ensembles to out-perform the oracle case, without relying on test domain labels. We suggest our approach, reported for domain adaptation, is broadly useful for NMT ensembling.

This chapter concludes the portion of this thesis exploring new approaches to NMT model fine-tuning and domain adaptation generally. The remaining chapters of original work apply domain adaptation techniques to problems not typically treated as related to domain-adaptation.

Chapter 7

Case study: Different sentence representations in NMT as complementary domains

This chapter draws from the following publications: Saunders, Feely, et al. (2020) in Sec. 7.2 and Saunders, Stahlberg, de Gispert, et al. (2018) in Sec. 7.3. Some work in Sec. 7.2 on proprietary data was performed during a research placement at SDL plc.

7.1 Motivation

In the preceding three chapters, we have developed effective measures for adapting NMT models to new domains, with a focus on good performance over multiple distinct domains. The approaches can be broadly split into techniques that vary the adaptation data, the adaptation procedure, or the inference procedure.

In this chapter we move away from strict notions of changing 'domain', instead presenting a case study on varying data representation (reviewed Sec. 2.1). We explore approaches that represent the same sentences in different ways, such as varying levels of subword or sub-character segmentation (Sec. 2.1.2) or added syntactic annotation (Sec. 2.1.3). In the thesis so far we have found that sets of models translating different domains have benefited from careful consideration of data selection, training and inference. In this chapter we address our fourth research question by showing that sets of models using different data representations can benefit from similar considerations.

In Sec. 7.2, we explore the impact of different representations choices at training and inference time for linguistically distant language pairs, focusing on sub-character representa-

tions for logographic source sentences. We show that simply changing a model's source data representation can lead to improvements in translation adequacy.

In Sec. 7.3, we address the assumption that all models in an inference-time ensemble must share a target language representation. Specifically we extend the idea of using target language syntactic annotations for NMT, developing a scheme to include such models in an ensemble with multiple target representations.

7.2 Sub-character language representations

Just as vocabulary, style or other sentence content can affect the quality of a translation or the convergence point of an NMT model trained on it, so too can the sentence's surface-level representation. For example, consider a model trained on sentences represented as word sequences versus one trained on the same sentences represented as character sequences. The character-based model might be more robust to spelling variation or novel inflections. The word-based model might produce more literal word-for-word translations. Models that encode or generate different surface-level representations of language are likely to provide different benefits to different sentences. We may therefore wish to use different models in different scenarios, as we often do for multi-domain NMT. As a result considerations made for multi-domain NMT, like choice of model or ensemble at inference time, may also be relevant here.

While a significant amount of existing work compares the benefits of character representations to word or subword language representations (Sec. 2.1.2), sub-character representations are less well studied. In this section we present original work exploring the strengths and weaknesses of sub-character representations, with a simple but novel inference-time scheme allowing NMT models to translate unseen logographic characters. While we interpret different surface-level text representations as behaving like domains, we do not adapt to them to translate given sentences, but explore their effects on translation quality for those sentences.

7.2.1 Sub-character decomposition for unseen characters

While Neural Machine Translation (NMT) has evolved rapidly in recent years, not all of its successful techniques are equally applicable to all language pairs. A particular example is the representation and translation of unseen tokens, which do not appear in the training data. With techniques like BPE subword decomposition (Sennrich, Haddow, and Birch, 2016d), an unseen word in an alphabetic language can in the worst case be represented as a sequence of

Char	Meaning	Sub-characters	Semantic sub-character
森	Forest	木木木	木 (Tree)
鰯	Sardine	魚弱	魚 (Fish)
校	School	木交	木 (Tree)

Table 7.1 Some characters with sub-character decompositions given by CHISE. Not all decompositions or sub-characters convey the semantic meaning of the character.

characters. Since alphabetic languages usually have few unique characters, it is reasonable to assume that all of these 'back-off' characters will be present in the limited model vocabulary.

We focus instead on the translation of unseen logographic characters as used in Chinese and Japanese text into alphabetic languages, a task that remains a challenge for NMT. Logographic writing systems may have many thousands of logograms, each representing at least one word, morpheme or concept as well as conveying phonetic and prosodic information. Inevitably some characters will either not be present in the training data, or will be present but too rare to be included in the vocabulary.

If the model is required to translate a previously unseen character, it will usually be replaced with an `UNK` (unknown word) token. The most likely outcome is that it will be ignored by the translation model, which will instead rely on the context of the unseen character to produce the translation. In the worst case, the presence of a previously-unseen character at inference time may harm the translation quality. This is a particular concern for NMT in low-resource domains, where there are fewer training examples to provide useful lexical context for unseen character translation.

Many logographic characters share sub-character components[1], which can carry semantic or phonetic meaning (Table 7.1). An intuitive approach to the logogram sparsity problem in NLP uses sub-character decompositions in place of characters.

Prior work on using sub-character decompositions in NMT has focused on leveraging shared sub-characters to improve Chinese-Japanese translation (Zhang and Komachi, 2019; Zhang and Komachi, 2018). The typical approach in this work is to decompose all logograms and learn BPE vocabularies over sub-character sequences.

We identify two motivations for using sub-characters in logographic NMT:

1. Sharing vocabularies between languages with similar sub-character decompositions, as in Chinese-Japanese translation.

[1] As discussed in Sec. 2.1.2 214 sub-character units are considered to be non-decomposable radicals, which are defined as a block in Unicode as of version 3.0 (Consortium, 2000). Here we follow prior work in using shallower decompositions which can include non-radical sub-character units.

2. Representing unseen characters – those not appearing in the training data – in semantically meaningful ways.

Our hypothesis is that, while complete sub-character decomposition for *all* characters might be useful in case 1, only *some* characters benefit from only semantic elements of the decomposition in case 2. In this section our focus is case 2. Using the representation-domain analogy, we could say that sentences with unseen characters constitute a small domain. We propose that it is better to approach this 'domain' with a specific model or inference procedure, rather than attempt to use the same approach for all sentences regardless of unseen character content.

The novel contributions described in this section are as follows:

- We compare ideograph-based sub-character schemes for Chinese-to-English and Japanese-to-English NMT with a strong BPE subword baseline, for both high- and low-resource domain translation.
- We evaluate both on general test sets, and on challenge sets which we construct such that all sentences have at least one character that was not seen in the training data.
- We demonstrate that, counter-intuitively, training models with indiscriminate sub-character decomposition can harm unseen character translation. Such models also give inconsistent performance on sentences with no unseen characters.
- We instead propose a set of extremely straightforward inference-time sub-character decomposition schemes requiring no additional models or training.

Sub-character decomposition schemes

Over 80% of Chinese characters can be broken down into both a semantic and a phonetic component (Liu, Chung, et al., 2010). The semantic meaning of a Chinese character often corresponds to the sub-character occupying its top or left position (Hoosain, 1991). These may be (but are not always) radicals: sub-characters that cannot be broken down any further. However, even top- or left-position radicals are not necessarily directly meaningful, as demonstrated in Table 7.1. Radical 魚 ('fish') has an intuitive semantic relationship with the character 鰯 ('sardine'), but the semantic connection of radical 木 ('tree') to character 校 ('school') is more abstract. The phonetic component is less likely to be helpful for translation to a non-logographic language, except in the case of transliterations.

Training with sub-character decomposition

We first explore the impact of two variations on ideograph-based sub-character decomposition applied to all characters in the source language. Following Zhang and Komachi (2019) we use decomposition information from the CHISE project[2], which provides ideograph sequences for CJK (Chinese-Japanese-Korean) characters. As well as ideographs, the sequences include ideographic description characters (IDCs), which convey the structure of an ideograph. While Zhang and Komachi (2019) use IDC information for Chinese-Japanese translation, use of structural sub-character information has not yet been explored for NMT to an alphabetic language.

IDCs may convey useful information about which sub-character component is likely to be the semantic or phonetic component, but they also make character representations significantly longer. We therefore compare training with sub-character decompositions with and without the IDCs.

Inference-only sub-character decomposition

Applying sub-character decomposition to all characters for training decreases the vocabulary size, but significantly lengthens source sequences. Additionally, all source characters are decomposed regardless of whether they might benefit from decomposition. We propose an alternative approach which applies sub-character decomposition only to unseen characters at inference time.

We apply decomposition if a test source sentence 1) contains an unseen character which 2) can be decomposed into at least one sub-character that is already present in the vocabulary. We do not include the entire decomposition, but keep only the sub-characters already in the model vocabulary. We experiment with both keeping all in-vocabulary sub-characters, and keeping only the leftmost (L) in-vocabulary sub-character from the break-down. We consider the left-only approach to be a reasonable heuristic: the breakdown produces radicals top-to-bottom and left-to-right, meaning the left-most radical will either be the top component or the left component, which is frequently the semantic component (Hoosain, 1991).

The inference-only decomposition approach has several advantages over training with sub-character decomposition. It is extremely fast, since decomposition is a pre-processing step before inference. It does not require training from scratch with very long sequences, which can harm overall performance. Sentences without unseen characters, which are unlikely to benefit from decomposition, are left completely unchanged by the scheme.

[2]Accessed via https://github.com/cjkvi/cjkvi-ids

Finally, the scheme is very flexible: decomposition can be applied to individual unseen characters on a case-by-case basis if necessary. For example, the presence of the 魚 ('fish') radical on the left of a character very often indicates that the character is for a type of fish, so applying inference-only decomposition to such characters will improve adequacy. Characters can in principle be excluded from decomposition if they do not benefit from it.

We convert some sub-character components to their base forms to improve character coverage. A small number of components change form in some cases. For example, 水 ('water'), can exist as its own character or as a radical, but often becomes 氵 when used on the left hand side of a character (e.g. 池, 'pond'). We manually define 30 such cases for inference-only decomposition, swapping the changed radical (unlikely to be in the vocabulary) for its base form (often in the vocabulary). This is unneeded when training with sub-character decomposition as all forms can be included in the vocabulary.

Even when radicals are replaced with their base form, not all radicals will be present in the vocabulary of a non-sub-character model. To address this problem we propose replacing the out-of-vocabulary radical with an in-vocabulary, non-radical character that conveys a related semantic meaning. Experimentally, we attempt this with a single character for both Chinese and Japanese, replacing radical 疒 ('illness'), which is not in the vocabulary, with character 病 ('illness').

Finally, a very simple approach to unseen sub-characters is to remove them from source sentences. This makes it unlikely that the character will be correctly translated, but saves the model from translating an `UNK`. We only apply this to characters which could be decomposed, so `UNK`s may still occur.

Examples of real sub-character decompositions for all schemes used in this work are shown in Table 7.2. Note that only the sub-characters already present in the vocabulary are included in inference-time decomposition, but for training decomposition all sub-characters are likely to appear in the vocabulary.

7.2.2 Experimental setup

Data

For both Chinese-English and Japanese-English, we first train a baseline model on a larger corpus and then adapt the same model to a smaller corpus. This lets us evaluate unseen character translation in both higher- and lower-resource settings. In both cases we evaluate on a corresponding standard test set where available, as well as an unseen characters test set. The latter is constructed from training sentences containing at least one decomposable logographic character otherwise not appearing in the training set. These sentences are held

Decomposition	鰯	瘡
Baseline	UNK	UNK
Training decompose	魚弱	疒倉
Training decompose (IDC)	⿰魚弱	⿰疒倉
Inference-only remove		
Inference-only decompose	魚弱	倉
Inference-only decompose (L)	魚	倉
Inference-only decompose (replace unseen radical)	魚弱	病倉
Inference-only decompose (L, replace unseen radical)	魚	病

Table 7.2 Training and inference-only decompositions used in this work for two characters. 鰯 ('sardine') has in-vocabulary semantic component 魚 ('fish') and 瘡 ('sores') has out-of-vocabulary semantic component 疒 ('illness'). Since sub-character 疒is not in the vocabulary, it does not appear in inference-only decomposition unless swapped with an in-vocabulary character e.g. 病 ('illness').

Set	**Chinese-English**		**Japanese-English**	
Training data source	Proprietary	CAS	ASPEC	KFTT
Train	50M	3M	2M	330K
General test set	2000	3981	1812	1160
Unseen chars test set	2140	1360	336	2243

Table 7.3 Sentence counts for Chinese-English and Japanese-English training and test sets. Chinese-English proprietary and CAS training corpora have no standard test sets, so we use the WMT news task WMT19 and WMT18 test sets respectively. The 'unseen chars' test sets are held out from the corresponding training sets such that every sentence has at least one unseen decomposable logographic character.

out from the the training data, so any logographic characters appearing only in an 'unseen chars' set are not seen at all during training.

To construct the unseen character set for the higher-resource domain we hold out training sentences with logographic characters appearing infrequently[3] in the whole corpus, then filter for source/target sentence length ratio < 3.5. We build the BPE vocabularies (Sennrich, Haddow, and Birch, 2016d) on the high-resource domain training set. The baseline source and all target BPE vocabularies consist of character sequences, while the sub-character BPE vocabularies consist of sub-character sequences, following Zhang and Komachi (2018). For the lower-resource domains the unseen sets are held-out sentences containing logographic characters not in the baseline source vocabulary, filtered as before.

For Chinese-English our baseline model is trained on a parallel training data set that is proprietary to SDL plc. This set contains web-crawled data from a mix of domains – we refer to it simply as the 'proprietary' dataset. We learn separate Chinese and English BPE vocabularies on this corpus with 50K merges. For the lower-resource-domain model we adapt to 3M sentence pairs from publicly available corpora made available by the Chinese Academy of Sciences (CAS)[4]. Since neither of these training sets have standard test set splits, we use the WMT news tasks test sets WMT19 and WMT18 zh-en for general evaluation of the higher- and lower-resource cases respectively (Barrault et al., 2019). WMT19 contains only seen characters, as do all but 2 lines of WMT18.

For Japanese-English, we train the higher-resource model on 2M scientific domain sentence pairs from the ASPEC corpus (Nakazawa et al., 2016). We learn separate Japanese and English BPE vocabularies on this corpus with 30K merges. Our smaller domain is the Kyoto Free Translation Task (KFTT) corpus (Neubig, 2011). We use the standard test sets for general evaluation. In the ASPEC test set 36 (2%) sentences contain unseen decomposable characters, as well as 180 (15.5%) sentences in the KFTT test set.

Model, training and inference

Our NMT models are all Transformer models with hyperparameters according Tensor2Tensor's base setting and a batch size of 4K tokens. For the higher resource domain models we train for 300K steps for Chinese-English and for 240K steps for Japanese-English. For the lower resource domains we fine-tune the trained models for 30K and 10K steps respectively.

We conduct inference via beam search with beam size 4. For ASPEC evaluation we evaluate Moses tokenized English with the Moses multi-bleu tool to correspond to the official

[3]No more than two occurrences for Chinese or three for Japanese, since Japanese has smaller datasets.
[4]Casia2015 and Casict2015 corpora from http://nlp.nju.edu.cn/cwmt-wmt/

Decomposition (training)	**Chinese-English**				**Japanese-English**			
	Higher-resource		**Lower-resource**		**Higher-resource**		**Lower-resource**	
	WMT19	Unseen	WMT18	Unseen	ASPEC	Unseen	KFTT	Unseen
None (Baseline)	**25.2**	**22.6**	**18.3**	**12.4**	**28.3**	13.5	**16.9**	**13.3**
Decompose	24.9	**22.6**	17.5	11.4	26.9	**14.8**	16.2	12.5
Decompose IDC	24.8	22.5	18.2	**12.4**	26.4	14.7	16.2	12.4

Table 7.4 BLEU scores for training with different decomposition schemes for higher- and lower-resource test sets. The baseline has no sub-character decomposition. Sub-character decomposition during training fails to improve general translation, and only improves unseen set translation for ASPEC, for which it also harms general translation.

WAT evaluation[5]. For all other results we report detokenized English using SacreBLEU. All BLEU is for truecased English.

7.2.3 Experiments on the impact of sub-character representations for unseen characters

We have two requirements when using sub-character decomposition for unseen character translation:

- Sets with few unseen characters (all general test sets except KFTT) should not experience performance degradation in terms of BLEU.
- Translation performance on unseen characters should improve.

Unseen character translation improvement may not be detectable by BLEU score, since the unseen character sets may only have one or two unseen characters per sentence. Moreover generating a hypernym, such as 'fish' instead of 'sardine' for 鰯, would not improve BLEU, despite being a more adequate translation than `UNK` and a more correct translation than e.g. 'salmon'. Consequently we also give translation examples for the most promising schemes at training- and inference-time.

Training with decomposition harms general translation

In Table 7.4 we give results after training with sub-character decomposition schemes. We start with a strong BPE baseline, and compare decomposition with and without the IDC structural information described above. On general test sets, we see BLEU degradation compared to the baseline, especially for Japanese-English. We note that our Japanese-English

[5]https://github.com/moses-smt/mosesdecoder/

Decomposition (inference)	Chinese-English				Japanese-English			
	Higher-resource		Lower-resource		Higher-resource		Lower-resource	
	WMT19	Unseen	WMT18	Unseen	ASPEC	Unseen	KFTT	Unseen
None (Baseline)	25.2	**22.6**	18.3	**12.4**	28.3	13.5	**16.9**	13.3
Remove unseen	25.2	**23.0**	18.3	**12.4**	**28.4**	**15.0**	16.8	**13.5**
Decompose unseen	25.2	22.7	18.3	11.8	28.3	14.2	16.7	12.9
Decompose unseen (L only)	25.2	**23.0**	18.3	12.0	**28.4**	14.6	16.7	13.3

Table 7.5 Higher- and lower-resource test set BLEU scores for the baseline models of Table 7.4 with different inference-time decomposition methods. Line 1 is duplicated from Table 7.4. Inference-time decomposition performs about the same as the baseline on general test sets, and some unseen sets see BLEU improvement.

ASPEC decomposed-training score is similar to the result for the same set achieved by Zhang and Komachi (2018) with ideograph decomposition. However, our non-decomposed baseline is much stronger, and so we are not able to replicate their finding that training with sub-character decomposition is beneficial to NMT from logographic languages to English. We suggest this degradation may be the result of training and inference with much longer sequences, which are well-established as challenging for NMT (Koehn and Knowles, 2017).

Interestingly we find that adding IDCs, which lengthen sequences, performs slightly better for the lower-resource cases, especially for Chinese-English. A possible explanation is that the longer sequences regularize adaptation in these cases, avoiding over-fitting to the highly specific lower-resource domains. However, these cases still show degradation relative to the baseline.

On the unseen sets, training with sub-character decomposition outperforms the baseline in terms of BLEU for the ASPEC unseen set. However, this is not a consistent result, with the baseline performing best or joint best in all other cases.

Inference-only decomposition leaves seen character translation unchanged, and may improve unseen character translation

Table 7.5 gives results for our inference-only unseen character decomposition schemes, compared to the baseline with no decomposition. Inference time decomposition has no effect on the Chinese-English test sets with no unseen characters. This is as we expect, since these test sets are unchanged. For Japanese-English a slight decrease on the KFTT general set (about 15% sentences with unseen characters) is balanced by a small improvement on the ASPEC general set (2% sentences with unseen characters). These results give a strong

advantage compared to training decomposition, which must be applied to all sentences whether they benefit or not, degrading performance in the case of Japanese-English.

Test sets with many unseen characters have a range of BLEU performance under inference-time decomposition. One consistent result is that left-only decomposition gives better scores than using all sub-characters. This may be explained by the fact that representing a character as multiple sub-characters may lead the model to generate a separate translation for each sub-character, harming performance. By contrast the leftmost sub-character tends to be the semantic component so may give good translation performance alone.

As a precision-based metric, BLEU is not an ideal measure of improving unseen character translation. Any such improvements under decomposition are more likely to improve adequacy than precision, since they often involve introducing synonyms or hypernyms. This difficulty is highlighted by the strong performance of the 'remove unseen' scheme which simply deletes unseen decomposable characters from source sentences. Clearly, such a scheme cannot improve the translation of these characters, although it may reduce the number of hypothesis tokens, inadvertently improving precision and therefore BLEU.

The higher performance of the decompose (L) scheme is more promising, since this is likely to actually generate translations for unseen characters. On a similar note, replacing the unseen 'illness' radical with a character conveying the same semantic meaning as described at the end of Sec. 7.2.1 does not affect BLEU for any set, but we do see noticeable improvements in adequacy for the handful of affected sentences.

Qualitative evaluation: inference-only decomposition can improve unseen character translation

We provide example translations under different training and inference decomposition schemes in Table 7.6. We observe some interesting differences in adequacy between training decomposition and inference-only decomposition. In particular, both Japanese translations with training decomposition feature a plausible but incorrect translation. With inference-only decomposition the translation is less fluent, but more generic and consequently more correct.

We note that training with sub-character decomposition has an unfortunate tendency to translate over-specific terms from spurious sub-character matches. For example, in the first (ASPEC) Japanese example, 魚 ('fish') is also the radical in 鮭 ('salmon'), and in the second (KFTT) Japanese example, 倉 ('storehouse') is also a major component in 槍 ('spear'). The model trained with sub-character decompositions therefore produces 'salmon' and 'spear' instead of 'sardine' and 'measles'. Meanwhile the inference-only left-radical heuristic produces 'fish' and 'disease', both of which are correct translations, if not reference-matching.

Chinese source (CAS)	飞肉切薄片，用蛋清糊上浆，下开水锅[汆] 透捞出。
English reference	Cut the wild chicken meat into thin slices, smear with egg white, [scald] thoroughly.
Baseline	Fleshy slice, slurry with egg whites, and get out of the boiling water pan.
Training decompose	Fly to cut sliver pieces of meat, slurp them with purine paste, and pick them up from the open water pan.
Inference decompose	Cut thin slices of meat, slurry with egg whites, get out of the boiling water pan.
Japanese source (ASPEC)	空気中では[鰯]油が最も酸化されやすく，ついで亜麻仁油，大豆油の順であった。
English reference	Due to its high contents of DHA and EPA, [sardine] oil FFA was most rapidly oxidized in air, followed by linseed and soybean oil FFAs.
Baseline	In the air, the sate oil was most easily oxidized, followed by linseed oil and soybean oil.
Training decompose	In air, salmon oil was most susceptible to oxidation, followed by linseed oil and soybean oil.
Inference decompose (L)	Fish oil was most oxidized in air, followed by linseed oil and soybean oil.
Japanese source (KFTT)	康元元年(1256 年) 赤斑[瘡]により死去。
English reference	In 1256, he died from [measles].
Baseline	In 1256, he died of a red spot.
Training decompose	In 1256, he died of a spear.
Inference decompose (L)	In 1256, he died from a red spot storehouse.
Inference decompose (L, replace radical)	In 1256, he died from a red spot disease.

Table 7.6 Examples of translation with different decomposition schemes from each of the three unseen sets extracted from publicly available corpora. We compare the most consistent training decomposition (no IDCs) and inference-only left-only (L) decomposition to the baseline. In the final Japanese example, we additionally compare swapping the unseen radical with an in-vocabulary character. Unseen characters and (approximate) reference translations are marked in square brackets.

We identify this pattern throughout the unseen-character sets for certain characters in particular. Characters for concrete nouns, such as types of fish, illness, bird, tree, and so on tend to be well-handled by inference-only decomposition with the left-sub-character heuristic and failed by the training decomposition scheme.

More abstract characters are more challenging for both schemes, such as those with radical 忄 ('heart') which often refer to an emotion. However, a major benefit of our approach is its flexibility; such poorly-handled characters could simply be excluded from the decomposition scheme, or replaced with a more appropriate non-radical character as we do for the 'illness' radical 疒. Future work on this problem could involve determining the most relevant sub-character component of an character, if any, rather than the simple left-only heuristic.

7.2.4 Sub-character decomposition summary

We explore the effect of sub-character decomposition on NMT from logographic languages into English. Decomposition for training may hurt general translation performance without necessarily helping unseen character translation. A domain adaptation analogy would consider a handful of challenging sentences, adapt a model to a set of spuriously connected training examples, and then use that model to translate *all* sentences, challenging or otherwise.

We instead propose a flexible inference-time sub-character decomposition procedure which targets unseen characters. We show that our scheme aids adequacy and reduces misleading overly-specific translation in unseen character translation. The scheme is straightforward, requires no additional models or training, and has no negative impact on sentences without unseen characters. Continuing the domain adaptation analogy, we can say we treat sentences with unseen characters as a distinct domain, and therefore treat them differently from other sentences at inference time.

7.3 Multi-representation ensembles for syntax-based NMT

In the previous section we demonstrated that the same sentence with a different representation can be treated like a different domain. In this section we extend this idea to multiple-representation ensembles. In Chapter 6 we showed that an ensemble of models from different domains can combine the benefits of knowledge from component domains. Here we explore ensembles of models with different target representations which benefit from the different representations.

Previous work has observed that NMT models trained to generate target syntax can exhibit improved sentence structure (Aharoni and Goldberg, 2017; Eriguchi et al., 2017) relative to those trained on plain-text, while plain-text models produce shorter sequences and so may encode lexical information more easily (Nadejde et al., 2017). In other words, syntactically annotated sentences and plain-text sentences can behave as complementary domains.

We hypothesize that an NMT ensemble would be strengthened if its component models were complementary in this way. However, ensembling typically requires component models to make predictions relating to the same output sequence position at each time step. Models producing different sentence representations must necessarily be synchronized to enable this.

In this section we first discuss practical considerations for representing target language syntax in NMT. Specifically, we discuss schemes for including English syntactic information in target sequences for Japanese-to-English translation. We then continue by developing a formalism for ensembles containing models with multiple target representations. We finally evaluate inference schemes with ensembles of models generating different syntax representations, or 'plain-text' sentences with no additional syntactic tags.

7.3.1 NMT with target syntax

Very long sequences are known to pose a challenge for NMT (Koehn and Knowles, 2017). We observed this effect in the previous section when including structural IDC elements in sub-character decompositions. Including syntactic annotations, such as POS tags or elements denoting the structure of a constituency tree, also significantly increases the length of a target sequence. Table 7.7 gives examples of different possible syntactic annotations with average number of tokens per sentence for ASPEC English training sentences. Adding a POS tag for each word in the sequence approximately doubles the number of tokens needed to represent the sentence[6]. A linearized constituency tree representation contains structural information as well as POS tags and may be far longer.

We therefore propose a derivation-based representation which contains the same structural information as the linearized constituency tree, but is more compact. A derivation as in line 4 of Table 7.7) represents the constituency tree as a sequence of generation rules obtained via a left-first traversal of the tree. This form still contains a great deal of repeated information, as every non-terminal appears initially on the right-hand side of a rule and later as the left-hand side of a rule. We instead define a linearized derivation consisting of the right-hand side of each rule. An end-of-rule marker, </R>, indicates the final non-terminal in each rule.

[6]In general sentence lengths do not quite double because each word has a single POS tag but may be represented by multiple subwords.

	Representation	Sample	Mean length
1	Plain-text	No complications occurred	27.5
2	POS/plain-text	DT No NNS complications VBD occurred	53.3
3	Linearized tree	(ROOT (S (NP (DT No) (NNS complications)) (VP (VBD occurred))))	120.0
4	Derivation	ROOT→S ; S→NP VP ; NP→DT NNS ; DT→No ; NNS→complications ; VP→VBD ; VBD→occurred	-
5	Linearized derivation	S</R> NP VP</R> DT NNS</R> No complications VBD</R> occurred	73.8

Table 7.7 Examples for proposed representations. Lengths are for the first 1M ASPEC English training sentences with BPE subwords (Sennrich, Haddow, and Birch, 2016d).

In this form every non-terminal appears exactly once. The original tree can be directly reproduced from the linearized derivation sequence, so structural information is maintained. The linearized derivation is still significantly longer than simply alternating POS tags with plain-text, since it contains many additional non-terminals. However, it conveys the same information as a linearized tree while being on average less than two thirds as long.

7.3.2 Ensembles with multiple target representations

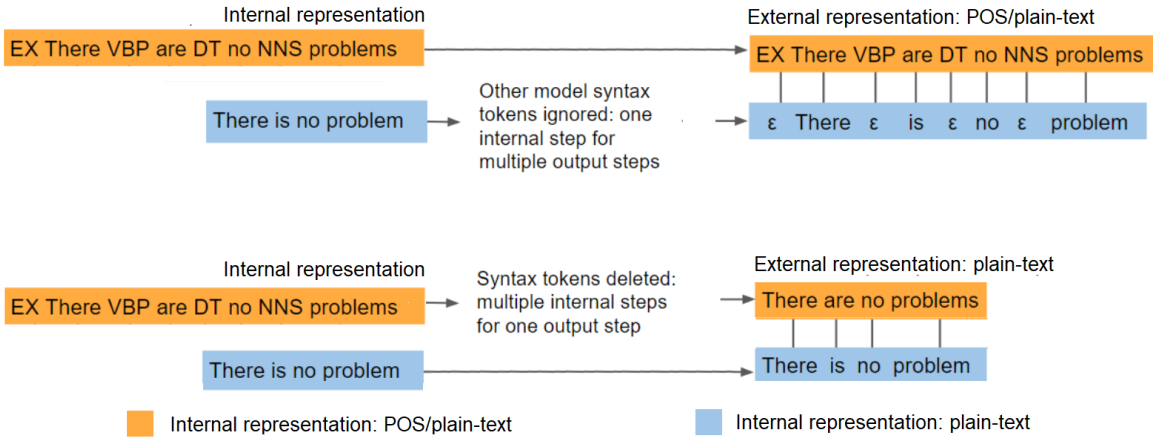


Fig. 7.1 Multiple models in an ensemble may have different internal representations, but the ensemble as a whole produces a single external representation. Internal representations can be converted to the external representation, allowing synchronized inference with multi-representation ensembles. Ignored tokens are indicated by ε.

Table 7.7 shows several different representations of the same hypothesis. We describe ensembles containing models with different target sentence representations in terms of *internal representations* and *external representations*. The ensembling decoder will produce a

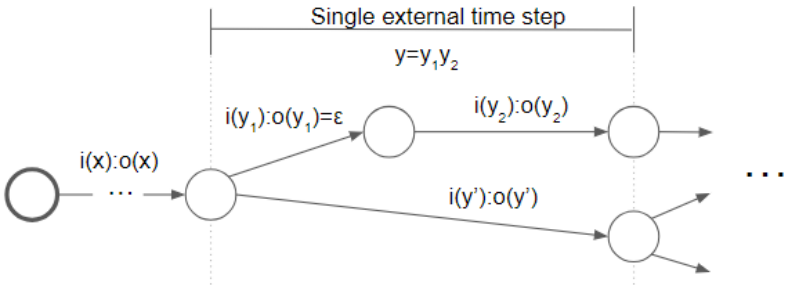


Fig. 7.2 Transducer mapping internal to external representations. A partial hypothesis might be $\mathsf{o}(\mathsf{xy}_2)$ in the external representation and $\mathsf{i}(\mathsf{xy}_1\mathsf{y}_2)$ in the internal representation.

single best hypothesis, or a list of the N best hypotheses, following the external representation. The models constituting the ensemble each have their own internal representation, which may or may not match the external representation. Examples are given in Figure 7.1 for ensembles of two models, one with a POS/plain-text internal representation and one with a plain-text internal representation.

To formulate an ensembling decoder over pairs of these representations, we assume we have a transducer T that maps from internal to external representation. Let $\mathscr{P}$ be the paths in T leading from the start state to any final state. A path $p \in \mathscr{P}$ maps internal representation $i(p)$ to external representation $o(p)$.

We formalize the multi-representation ensemble for two models with different representations. We can do this without loss of generality since we only need to synchronize pairs of internal and external representations. The synchronization could if necessary be defined via a different T for each internal/external representation pair in the ensemble if it contained more than two models.

We therefore assume that two NMT systems are trained, one using the internal representation and one using the external representation, giving models P_i and P_o which we wish to ensemble[7]. An ideal equal-weight ensembling of P_i and P_o yields

$$p^* = \underset{p \in \mathscr{P}}{\operatorname{argmax}}\, P_i(i(p))\, P_o(o(p)) \tag{7.1}$$

with $o(p^*)$ as the external representation of the translation.

In practice, beam decoding is performed in the external representation, i.e. over projections of paths in $\mathscr{P}$ [8]. Let $h = h_1 \ldots h_j$ be a partial hypothesis in the external representation,

[7]For completeness we note that the external representation of the ensemble does not strictly need to match the internal representation of any component model. However, this introduces unnecessary complexity and has no benefit, since no constituent model would then correctly score the external representation.

[8]See the `tokenization` wrappers in https://github.com/ucam-smt/sgnmt

consisting of a path prefix x and a 'current' output token $h_j = o(y)$. The set of partial paths yielding h are:

$$M(h) = \{(x,y)|xyz \in \mathscr{P}, o(x) = h_{<j}, o(xy) = h\} \tag{7.2}$$

Here z is the path suffix. In other words, it must be possible to reach the end state in the external representation via some $z \in \mathscr{P}$. The ensembled score of h is then:

$$P(h_j|h_{<j}) = P_o(h_j|h_{<j}) \times \max_{(x,y)\in M(h)} P_i(i(y)|i(x)) \tag{7.3}$$

The max performed for each partial hypothesis h is itself approximated by a beam search. This leads to an outer beam search over external representations with inner beam searches for the best matching internal representations. As search proceeds, each model score is updated separately with its appropriate representation. Symbols in the internal representation are consumed as needed to stay synchronized with the external representation, as illustrated in Figure 7.2; epsilons are consumed with a probability of 1.

The complexity of the transduction depends on the representations. For example, mapping from word to BPE representations is straightforward, and mapping from a linearized syntax representation to plain-text simply deletes syntax tokens.

7.3.3 Experimental setup

Data

We report all experiments for Japanese-to-English translation. Our training data is the first 1M training sentences of the Japanese-English ASPEC data (Nakazawa et al., 2016). The ASPEC training set is sorted by alignment quality, and the first 1M sentence pairs are much cleaner than the subsequent 1M; we note that this results in different scores compared to the Japanese-English ASPEC baseline in the previous section, where a major objective was character coverage.

All models use plain-text BPE Japanese source sentences. English constituency trees are obtained using CKYlark (Oda et al., 2015), with words replaced by BPE subwords. We train separate Japanese (lowercased) and English (cased) BPE vocabularies on the plain-text, with 30K merges each. Non-terminals, including POS tags, are included as separate tokens. The linearized derivation uses additional tokens for non-terminals with </R>. As we wish to compare different target representations we train models with representations 1, 2, 3 and 5 shown in Table 7.7.

Model, training and inference

We primarily experiment using Transformer models. All Transformer architectures are Tensor2Tensor's base Transformer model (Vaswani, Bengio, et al., 2018) with a batch size of 4096 tokens. In all cases we decode using SGNMT (Stahlberg, Hasler, Saunders, et al., 2017) with beam size 4, using checkpoint averaging over the final 20 checkpoints.

For comparison with earlier target syntax work, we also train two RNN attention-based seq2seq models (Bahdanau, Cho, et al., 2015) with normal SGD to produce plain-text BPE sequences and linearized derivations. For these models we use embedding size 400, a single BiLSTM layer of size 750, and a batch size of 80 sequences.

We decode with individual models and two-model ensembles, comparing results for single-representation and multi-representation ensembles. Each multi-representation ensemble consists of the plain-text BPE model and one other individual model. We report case-sensitive BLEU for tokenized English using the multi-bleu tool to correspond to prior work reported in official WAT evaluations.

7.3.4 Experiments on ensembles with multiple target representations

Syntactic representations benefit from large batch sizes

Syntactic representations involve much longer sequence lengths than plain-text representations. Our NMT framework defines batch size as total source and target tokens per batch, so a syntactic model will 'see' fewer training sentence pairs per mini-batch gradient update. During NMT training, by default, the gradients used to update model parameters are calculated over individual mini-batches. A possible consequence is that batches with fewer sequences per update may have 'noisier' estimated gradients than those with more sequences. We therefore first investigate the susceptibility of syntax and plain-text representations to training difficulties when varying batch size.

Simply increasing the batch size can potentially improve convergence while requiring fewer model updates (Smith et al., 2018). However, with large batches the model size may exceed available GPU memory. Training on multiple GPUs is another way to increase the amount of data used to estimate gradients, but requires significant resources. Instead we avoid the problem by using delayed SGD updates. We accumulate gradients over a fixed number of batches before using the accumulated gradients to update the model. This lets us effectively use very large batch sizes without requiring multiple GPUs.

We experiment with both delayed SGD updates and reducing the learning rate. The results in Table 7.8 show that large batch training via delayed SGD updates can significantly improve the translation performance of single Transformers models. The improvement is

Representation	Batches / update	Learning rate	Test BLEU
Plain-text	1	0.025	27.5
	1	0.2	27.2
	8	0.2	28.9
Linearized derivation	1	0.025	25.6
	1	0.2	25.6
	8	0.2	28.7

Table 7.8 Japanese-English ASPEC test set BLEU for single Transformer models with plain-text and linearized derivation representations. Models are trained to convergence on 1M ASPEC training sentences for batch size 4096 tokens.

particularly evident for the linearized derivation model, which produces longer sequences. Accumulating the gradient over 8 batches of size 4096 gives a 1.7 BLEU improvement for the plain-text model and 3.1 BLEU improvement for the linearized derivation model. While Smith et al. (2018) suggest that decaying the learning rate can have a similar effect to large batch training, we find that reducing the initial learning rate by a factor of 8 alone does not give the same improvements.

We suggest that large batch training may be necessary for NMT with syntactically annotated sentence representations to perform on par with plain-text NMT, possibly due to increased gradient estimate noise. However, using 'effective' large batches with delayed SGD updates appears to be sufficient. We use delayed SGD with 8 batches per gradient update to train all remaining models in this section.

Multi-representation ensembles can improve syntax-based NMT

We proceed to compare individual NMT models trained on either plain-text or syntactically-annotated sentence representations. Results are shown in Table 7.9. For comparison with prior work on syntax for NMT, we first experiment with RNN-based models. We find that RNN-based syntax models can equal plain-text RNN models as in Aharoni and Goldberg (2017). Eriguchi et al. (2017) find that a translation model which also performs dependency parsing achieves a 1 BLEU improvement on the same ASPEC test set, but over a much weaker baseline.

Our plain-text Transformer baseline is very strong, outperforming the best listed system on ASPEC Ja-En at time of experiments (an 8-model ensemble (Morishita, Suzuki, et al., 2017)) as well as the best listed model of comparable size at time of writing (a single Transformer base model trained on 3M sentences (Dabre et al., 2018)). Our syntax models achieve similar results despite producing much longer sequences. Table 7.8 indicates that large batch training is instrumental in this. Our plain-text models outperforms all syntax-

Architecture	Representation	Dev BLEU	Test BLEU
Seq2seq	Best WAT17 result (8-model ensemble) (Morishita, Suzuki, et al., 2017)	-	28.4
	Plain-text	21.6	21.2
	Linearized derivation	21.9	21.2
Transformer	Best listed WAT result to date[9]using single base Transformer (Dabre et al., 2018)	-	28.6
	Plain-text	28.0	28.9
	Linearized tree	28.2	28.4
	Linearized derivation	28.5	28.7
	POS/plain-text	28.5	29.1

Table 7.9 Single models on Ja-En. Contemporary evaluation results included for comparison.

Ensemble type	Representation		Test BLEU	Δ
Single-representation ensemble	Plain-text		29.2	0.3
	Linearized tree		28.6	0.2
	Linearized derivation		28.8	0.1
	POS/plain-text		29.3	0.2
	External representation	**Internal representation**		
Multi-representation ensemble	Linearized tree	Plain-text	28.9	0.0
	Plain-text	Linearized tree	28.7	-0.2
	Plain-text	Linearized derivation	28.8	-0.1
	Linearized derivation	Plain-text	**29.4**	**0.5**
	POS/plain-text	Plain-text	29.3	0.2
	Plain-text	POS/plain-text	29.4	0.3

Table 7.10 Ja-En Transformer ensembles. Column Δ gives test BLEU improvement over best component model in each ensemble.

based models except POS/plain-text. More compact syntax representations perform better, with POS/plain-text outperforming linearized derivations, which outperform linearized trees.

Finally, we explore multi-representation ensembles containing plain-text and syntactically-annotated sentence representations. Ensembles of two identical models trained with different seeds only slightly improve over the single model (Table 7.10). However, an ensemble of models producing plain-text and linearized derivations improves by 0.5 BLEU over the plain-text baseline.

We note that it is necessary to choose internal and external representation carefully, as an ensemble of the same two models with internal and external representation swapped performs slightly worse than either model individually. In general, a multi-representation ensemble

[1]http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/list.php?t=2&o=4

Reference	Low-energy electron *microscope* (LEEM) and photoelectron *microscope* (PEEM) *were attracted* attention as new surface electron *microscope*.
Plain-text	Low energy electron *microscope* (LEEM) and photoelectron *microscope* (PEEM) *are noticed* as new surface electron *microscope*.
Linearized derivation	Low-energy electron *microscopy* (LEEM) and photoelectron *microscopy* (PEEM) *are attracting* attention as new surface electron *microscopes*.

Table 7.11 Sample generated translations from individual models, detokenized, with differences *emphasized*. We note that the reference itself may be ungrammatical, as in this case. The linearized derivation model achieves verb tense agreement (present plural) and noun agreement ('as new ... microscopes'), unlike the other translations.

with a richer, syntax-augmented external representation tends to outperform an ensemble where syntax tokens are only produced internally.

By ensembling syntax and plain-text we hope to benefit from their complementary strengths. To highlight these, we examine hypotheses generated by the plain-text and linearized derivation models. We find that the syntax model can be more grammatical, even when the plain-text model may share more vocabulary with the reference (Table 7.11).

In ensembling plain-text with a syntax external representation we observed that in a small proportion of cases non-terminals were over-generated, due to the mismatch in target sequence lengths. Our solution was to penalize scores of non-terminals under the syntax model by a constant factor.

It is also possible to constrain decoding of linearized trees and derivations to well-formed outputs. However, we found that this gives little improvement in BLEU over unconstrained decoding although it may be an interesting line of research for applications making use of the generated parses.

7.3.5 Multi-representation ensembling summary

We report strong performance with individual models that improves over comparable publicized shared task model results on the ASPEC Ja-En test set. We train these models using a delayed SGD update training procedure that is especially effective for the long representations arising from including target language syntactic information in the output. We further improve on the individual results via multi-representation ensembles. This inference scheme allows ensembling of models producing different output representations, such as plain-text with subword units and syntax.

These techniques were originally primarily proposed as practical approaches to including target syntax in NMT. In the context of this thesis, they are also a further sign of the potential benefits of using diverse systems to conduct inference.

7.4 Conclusions

We view language representation as a form of domain with its own strengths and weaknesses for translating different kinds of language. In this chapter we demonstrate that these 'domains' can be mutually beneficial given a scheme that allows their combination at inference time.

We describe in Sec. 7.2 how simple adjustments to the data representation alone can significantly improve translation of unseen characters. By contrast, we find that extensive changes to the data involving retraining a model from scratch do not necessarily solve the coverage problem for logographic characters. Our unseen character 'domain'-specific approach out-performs the general-purpose modelling approach.

In Sec. 7.3 we develop a new formulation for ensembling that allows combination of predictions from models with multiple target representations. We find that these representations can be complementary when combined in an ensemble.

Chapter 8

Case study: Gender bias reduction as a domain adaptation problem

This chapter draws from Saunders and Byrne (2020b) throughout. Some aspects of Sec. 8.3 draw from my contributions to Tomalin et al. (2021). Some aspects of Sec. 8.1 and Sec. 8.5 draw from Saunders, Sallis, et al. (2020)

8.1 Motivation

This thesis has throughout described the advantages and pitfalls of domain-specific NMT. Systems may specialize in translating specific topics or genres by adapting or weighting towards language that occurs in a given domain. This can be thought of as a bias towards certain translations under certain circumstances. Removing these biases might improve generalization, but would not necessarily be advantageous when translating a specific domain (Shah et al., 2020).

However, problems arise when the model learns spurious correlations, especially those corresponding to human demographics. For example, an English-to-Spanish NMT model might learn that sentences containing the word 'doctor' typically involve masculine pronouns and referents and those containing 'nurse' involve feminine pronouns and referents. As a result, the model might always translate 'This is the doctor' into a sentence with a masculine inflected noun – 'Este es el médico' – and 'This is the nurse' into a sentence with a feminine inflected noun – 'Esta es la enfermera'.

As reviewed in Sec. 3.6, it has been recently demonstrated that NMT systems often exhibit such behaviour, which we term gender bias: behaviour which 'systematically and unfairly discriminate[s] against certain individuals or groups of individuals in favor of others'

(Friedman and Nissenbaum, 1996). Specifically, translation performance favours referents fitting into groups corresponding to social stereotypes, such as male doctors.

Such systems propagate harms to users and referents. Referents may experience erasure – for example, a non-male doctor or non-female nurse would be incorrectly gendered by the above translations. Systems may also cause allocational harms if the incorrect translations are used as inputs to other systems (Crawford, 2017). System users also experience representational harms via the reinforcement of stereotypes associating occupations with a particular gender (Abbasi et al., 2019). Even if they are not the referent, the user may not wish for their words to be translated in such a way that they appear to endorse social stereotypes. The user will also experience a lower quality of service in receiving grammatically incorrect translations. To summarize, it is desirable to avoid this behaviour if possible.

In this chapter we address our final research question from Sec. 1.1.1, discussing schemes for mitigating gender bias in NMT inspired by the connection between useful domain-specific bias and undesirable gender bias. We apply data selection, model adaptation and inference techniques from throughout this thesis. Our approaches achieve significant improvements with very little computational cost by treating gender bias in NMT as a domain adaptation problem.

In Sec. 8.2 we describe the evaluation framework and metrics used throughout this chapter for measuring the effects of gender bias on NMT. In Sec. 8.3 we describe data-centric approaches to mitigating gender bias, both re-training on adjusted datasets and adapting to synthetic or semi-synthetic data. In Sec 8.4 we address the problem of catastrophic forgetting when adapting to a small, synthetic dataset by applying regularized adaptation and constrained inference techniques. In Sec. 8.5 we explore the concept of gender 'domain' signals in more detail, introducing explicit source-language word-level gender tags.

8.2 Measuring gender bias in NMT

WinoMT (Stanovsky et al., 2019) is a challenge set for evaluating gender bias in NMT across language pairs when translating from English. It permits automatic bias evaluation for translation into ten target languages with grammatical gender. The source side of WinoMT is 3888 gender-labelled sentences from Winogender (Rudinger et al., 2018) and WinoBias (Zhao, Wang, et al., 2018). The structure of the WinoMT test set is given in Table 8.1.

The test set is a set of coreference resolution examples in which each sentence contains a primary entity which is co-referent with a pronoun, as well as secondary entity. An example is the first sentence in WinoMT:

The developer argued with the designer because she did not like the design.

Label	**WinoMT subset**		
	Full	Pro-stereotypical	Anti-stereotypical
Male	1826	792	794
Female	1822	792	790
Neutral	240	0	0

Table 8.1 Summary of sentence counts for different gender labels for WinoMT (Stanovsky et al., 2019)

The primary entity is 'the developer', which is co-referent with the pronoun 'she'. The translation of 'the developer' should therefore be feminine-inflected for languages that gender-inflect human-referent nouns. The secondary entity is 'the designer', which should be gendered according to linguistic conventions for human-referent nouns where gender is ambiguous, unless additional context is available. In the languages handled by WinoMT the default inflection for ambiguous cases is the masculine[1].

We note there is semantic ambiguity in some English sentences used in WinoMT (González et al., 2020). Perhaps in the above example the designer, specializing in design, is scornful of the developer's attempts. For the majority of this chapter we will nevertheless rely on WinoMT as a well-recognized and broadly applied test set for gender translation (Kocmi, Limisiewicz, et al., 2020). In Sec. 8.5 we also explore the effects of removing the semantic ambiguity with explicit gender tags.

WinoMT evaluation extracts the grammatical gender of the primary entity from each translation hypothesis by automatic word alignment followed by morphological analysis. WinoMT then compares the translated primary entity with the gold gender, with the objective being a correctly gendered translation.

The WinoMT test set has approximately equal numbers of male- and female- labelled sentences, with a small number of neutral-labelled sentences (Table 8.1) The authors emphasize the following metrics over the challenge set:

- **Accuracy** – percentage of hypotheses with the correctly gendered primary entity.
- $\boldsymbol{\Delta G}$ – difference in F_1 score between the set of sentences with masculine entities and the set with feminine entities.
- $\boldsymbol{\Delta S}$ – difference in accuracy between the set of sentences with pro-stereotypical ('pro') entities and those with anti-stereotypical ('anti') entities, as determined by Zhao, Wang, et al. (2018) using US labour statistics. For example, the 'pro' set contains male doctors and female nurses, while 'anti' contains female doctors and male nurses.

[1] Although this is not the case for all gender-inflected languages (Corbett et al., 1999).

Our main objective is increasing accuracy – the percentage of correctly inflected primary entities. We also report on ΔG and ΔS for ease of comparison to previous work. Ideally the absolute values of ΔG and ΔS should be close to 0. A high positive ΔG indicates that a model translates male entities better, while a high positive ΔS indicates that a model stereotypes male and female entities. Large negative values for ΔG and ΔS, indicating a bias towards female or anti-stereotypical translation, are as undesirable as large positive values.

We note that ΔS can be significantly skewed by low-accuracy systems. A model generating male forms for most test sentences, stereotypical roles or not, will have very low ΔS, since its pro- and anti-stereotypical class accuracy will both be about 50%.

We wish to reduce gender bias without reducing translation performance. WinoMT is designed to work without target language references, and so it is not possible to measure translation performance on this set by measures such as BLEU. We therefore report BLEU on separate, general test sets for each language pair.

8.3 Reducing the effects of gender bias in NMT by changing the training data

In this section we compare data-centric approaches to reducing the effects of gender bias in NMT models, comparing different datasets for retraining and adaptation purposes. Our hypothesis is that the absence of gender bias can be treated as a small domain for the purposes of NMT model adaptation. In this case adaptation to a well-formed small 'in-domain' dataset may give better results than attempts at removing bias-inducing sentences from the entire original dataset. This adaptation approach can be viewed as an attempt to 'unlearn' problematic behaviour in much the same way as our data-centric, filtering-based approaches aimed to reduce exposure bias effects in Sec. 4.3.

8.3.1 Datasets for training and adaptation

Complete manual 'debiasing' is infeasible. Human language is both complex and evolving, and the contexts in which different human populations interact with NLP tools are also subject to change, leading to the possibility of new, emergent biases (Bender and Friedman, 2018). We are therefore unable to predict all possible negative model behaviours or the data features that might trigger them. Consequently, work on negative model behaviour generally focuses on clearly-defined instances of undesirable language correlations, such as the profession-oriented gender bias assessed by WinoMT. Even considering these specific

subsets of language, a machine translation dataset typically contains millions of parallel sentence pairs, which would simply be impractical for manual human editing.

However, some level of masculine-feminine vocabulary occurrence or co-occurrence balancing can be automated. We explore this approach for reducing gender bias in bilingual training data prior to training as a contrast to creating a small, synthetic dataset for mitigating gender bias post-training.

Up- and down-sampling

Up-sampling and down-sampling the original dataset are intuitive strategies for simply changing the ratio of gendered terms in the training set. When down-sampling we automatically remove data until the overall counts of English masculine and feminine gendered terms are approximately equal. When up-sampling we automatically add duplicated data until the counts are approximately equal.

By 'gendered term' we include many pronouns, nouns and terms of address that possess gender-related connotations in English. We identify 'gendered sentences' as those containing at least one gendered antonym from the list used by Zhao, Wang, et al. (2018). The list consists of 104 English word-pairs where the words are gendered antonyms of each other (e.g., 'son/daughter', 'he/she', 'husband/wife')[2].

The schemes for down- and up-sampling the dataset were as follows:

- Iterate through the English side of all sentence pairs, counting the number of masculine and feminine gendered entities in each sentence.
- If down-sampling:
  - Add a sentence pair to the final dataset only if the English side has the same number of masculine and feminine entities.
- If up-sampling:
  - Include all gendered sentence pairs in the final dataset
  - Estimate the overall gender skew as the total number of masculine entities in all English sentences minus the total number of feminine entities
  - Continue to iterate through gendered sentence pairs, adding them to the final dataset again if they reduce the absolute overall skew.

[2]The stopword list and a 'gender-swapping' script are from the authors of Zhao, Wang, et al. (2018) at https://github.com/uclanlp/corefBias.

– Stop when overall skew reaches 0.

The down-sampling scheme described above ensures that every single batch of sentences has equal numbers of masculine and feminine gendered terms on the English side, which would not be the case if individually gender-skewed sentences were included in the dataset. It moreover usefully demonstrates a potential approach to data 'balancing' prior to training. Intuitive alternatives to the up-sampling scheme described above include a greedy approach, adding additional sentences in an order that maximally reduces the skew, and a cautious approach, adding only those sentences with minimal skew. The greedy approach adds fewer sentences, but individual sentences will necessarily be more skewed on the English side. The cautious approach results in significantly more up-sampling and hence data duplication. Aside from their questionable efficacy, exploring these alternatives is very expensive, since model retraining can take days for a high-resource language pair like English-German. We believe our randomized approach is a reasonable compromise.

Semi-synthetic counterfactual datasets

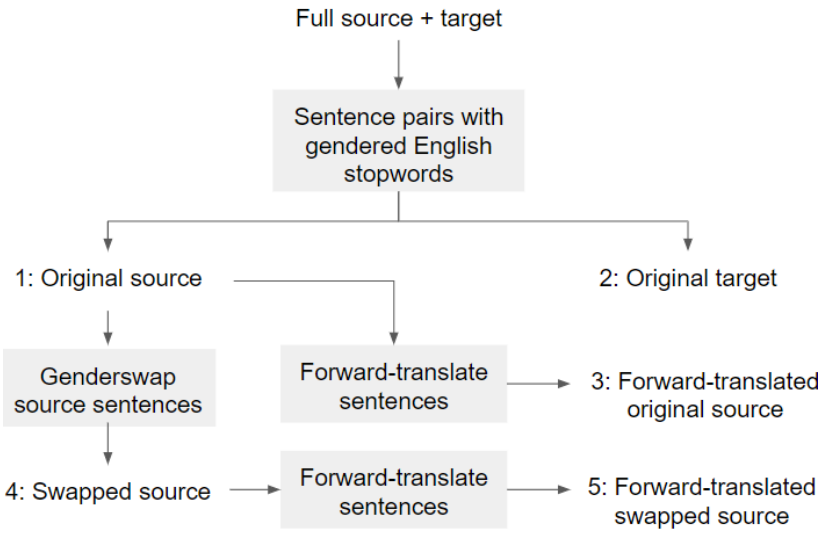


Fig. 8.1 Generating counterfactual datasets for adaptation. The **Original** set is 1||2, a simple subset of the full dataset. **FTrans original** is 1||3, **FTrans swapped** is 4||5, and **Balanced** is 1,4||2,5

For contrast, we describe an approximately counterfactual dataset for both retraining and fine-tuning. Counterfactual data augmentation is an intuitive solution to bias from data over-representation (Lu et al., 2020). It involves identifying the subset of sentences containing bias – in this case gendered terms – and, for each one, adding an equivalent sentence with the bias reversed – in this case a gender-swapped version.

Gender-swapping is relatively simple for English, which does not mark grammatical gender morphologically in articles or verbs, and only occasionally for nouns (e.g., 'actor' / 'actress'). However, the process is more complex for inflected languages where all the gendered parts-of-speech that occur in a sentence must be identified and updated. As well, gender-swapping translation training data requires that the same entities are swapped in the corresponding parallel sentence. A robust scheme for gender-swapping multiple entities in inflected language sentences directly, together with corresponding parallel text, would be a research project in itself[3]. Instead we suggest a rough but straightforward approach for counterfactual data augmentation for machine translation from English. To the best of our knowledge this is the first application of counterfactual data augmentation to parallel sentences.

We first perform simple gender-swapping on the subset of the English source sentences with gendered terms. We use the same set of gendered terms from Zhao, Wang, et al. (2018) as for up- and down-sampling, as well as their swapping script. We then greedily forward-translate the gender-swapped English sentences with a baseline NMT model trained on the the full source and target text, producing gender-swapped target language sentences.

This lets us compare four related sets for counterfactual data adaptation, as illustrated in Figure 8.1:

- **Original**: a subset of parallel sentences from the original training data where the source sentence contains gendered stopwords.
- **Forward-translated (FTrans) original**: the source side of the *original* set with forward-translated target sentences.
- **Forward-translated (FTrans) swapped**: the *original* source sentences are gender-swapped, then forward-translated to produce gender-swapped target sentences.
- **Balanced**: the concatenation of the *original* and *FTrans swapped* parallel datasets. This is twice the size of the other counterfactual sets.

Comparing performance in adaptation of *FTrans swapped* and *FTrans original* lets us distinguish between the effects of gender-swapping and of obtaining target sentences from forward-translation. We also include a comparison to using counterfactual data augmentation when training from scratch. In this case we simply shuffle *FTrans swapped* into the original dataset.

[3] See Zmigrod et al. (2019) for a discussion of some of the difficulties in monolingual gender-based data augmentation for languages with rich morphology.

Synthetic profession datasets

Finally, we construct a tiny, trivial set of synthetic sentences for fine-tuning with equal numbers of masculine and feminine entities. We first define English sentences which we can easily translate into each target language. The sentences follow the template:

The [*PROFESSION*] *finished* [*his*|*her*] *work.*

We refer to this as the *handcrafted* set. Each profession is from the list collected by Prates et al. (2019) from US labour statistics. We simplify this list by removing field-specific adjectives. For example, we have a single profession 'engineer', as opposed to specifying industrial engineer, locomotive engineer, etc. In total we select 194 professions, giving just 388 sentences in a gender-balanced set.

With manually translated masculine and feminine templates, we simply translate the masculine and feminine forms of each listed profession for each target language. In practice this translation is via an MT first-pass for speed, followed by manual checking, but given available lexicons this could be further automated. We note that the handcrafted sets contain no examples of coreference resolution and very little variety in terms of grammatical gender. In Sec. 8.5 we describe sets of more complex sentences targeted at the coreference task, as well as extensions to gender-neutral language.

We wish to distinguish between a model which improves gender translation, and one which improves its WinoMT scores simply by learning the vocabulary for previously unseen or uncommon professions. We therefore create a *handcrafted no-overlap* set, removing source sentences with professions occurring in WinoMT to leave 216 sentences. We increase this set back to 388 examples with balanced adjective-based sentences in the same pattern, e.g. *The tall* [*man*|*woman*] *finished* [*his*|*her*] *work.*

8.3.2 Experimental setup

Data

WinoMT provides an evaluation framework for translation from English to eight diverse languages. We select three pairs for experiments: English to German (en-de), English to Spanish (en-es) and English to Hebrew (en-he). Our selection covers three language groups with varying linguistic properties: Germanic, Romance and Semitic. Training data available for each language pair also varies in quantity and quality. We filter training data based on parallel sentence lengths and length ratios.

For **en-de**, we use 17.6M sentence pairs from WMT19 news task datasets (Barrault et al., 2019). We validate on newstest17 and test on newstest18.

For **en-es** we use 10M sentence pairs from the United Nations Parallel Corpus (Ziemski et al., 2016). While still a large set, the UNCorpus exhibits far less diversity than the en-de training data. We validate on newstest12 and test on newstest13.

For **en-he** we use 185K sentence pairs from the multilingual TED talks corpus (Cettolo, Niehues, Stüker, Bentivogli, and Federico, 2014). This is both a specialized domain and a much smaller training set. We validate on the IWSLT 2012 test set and test on IWSLT 2014.

Table 8.2 summarises the sizes of datasets used, including their proportion of gendered sentences and ratio of sentences in the English source data containing male and female stopwords. A gendered sentence contains at least one English gendered stopword as used by Zhao, Wang, et al. (2018).

Interestingly all three datasets have about the same proportion of gendered sentences: 11-12% of the overall set. While en-es appears to have a much more balanced gender ratio than the other pairs, examining the data shows this stems largely from sections of the UNCorpus containing phrases like 'empower women' and 'violence against women', rather than gender-balanced professional entities.

	Training	**Gendered training**	**M:F**	**Test**
en-de	17.5M	2.1M	2.4	3K
en-es	10M	1.1M	1.1	3K
en-he	185K	21.4K	1.8	1K

Table 8.2 Parallel sentence counts. A gendered sentence pair has minimum one gendered stopword on the English side. M:F is ratio of male vs female gendered training sentences.

For en-de and en-es we learn joint 32K BPE vocabularies on the training data (Sennrich, Haddow, and Birch, 2016d). For en-he we use separate source and target vocabularies. The Hebrew vocabulary is a 2K-merge BPE vocabulary, following the recommendations of Ding et al. (2019) for smaller vocabularies when translating into lower-resource languages. For the en-he source vocabulary we experimented both with learning a new 32K vocabulary and with reusing the joint BPE vocabulary trained on the largest set – en-de – which lets us initialize the en-he system with the pre-trained en-de model. The latter resulted in higher BLEU and faster training.

Model, training and inference

For all models we use a Transformer model with the 'base' parameter settings given in Tensor2Tensor. We train baselines to validation set BLEU convergence on one GPU, delaying gradient updates by factor 4 to simulate 4 GPUs (Saunders, Stahlberg, de Gispert, et al., 2018). During fine-tuning training is continued without learning rate resetting. Normal and

lattice-constrained decoding is via SGNMT with beam size 4. BLEU scores are calculated for case-sensitive, detokenized output using SacreBLEU.

8.3.3 Experiments in improving gender translation accuracy with data-centric methods

Baseline analysis

	en-de			en-es			en-he		
	Acc	ΔG	ΔS	Acc	ΔG	ΔS	Acc	ΔG	ΔS
Microsoft	**74.1**	**0.0**	30.2	47.3	36.8	23.2	48.1	14.9	32.9
Google	59.4	12.5	12.5	53.1	23.4	21.3	**53.7**	**7.9**	37.8
Amazon	62.4	12.9	16.7	**59.4**	**15.4**	22.3	50.5	10.3	47.3
SYSTRAN	48.6	34.5	**10.3**	45.6	46.3	15.0	46.6	20.5	**24.5**
Baseline	60.1	18.6	13.4	49.6	36.7	**2.0**	51.3	15.1	26.4

Table 8.3 WinoMT accuracy, masculine/feminine bias score ΔG and pro/anti stereotypical bias score ΔS for our baselines compared to commercial systems, whose scores are quoted directly from Stanovsky et al. (2019).

In Table 8.3 we compare our three baselines to commercial systems on WinoMT, using results quoted directly from Stanovsky et al. (2019). Our baselines achieve comparable accuracy, masculine/feminine bias score ΔG and pro/anti stereotypical bias score ΔS to four commercial translation systems, outscoring at least one system for each metric on each language pair.

The ΔS for our en-es baseline is surprisingly small. Investigation shows this model predicts male and female entities in a ratio of over 6:1. Since almost all entities are translated as male, pro- and anti-stereotypical class accuracy are both about 50%, making ΔS very small. This highlights the importance of considering ΔS in the context of other metrics, such as ΔG, which should be close to 0, and the ratio of M:F predictions, which should be 1.0 on the original WinoMT set.

Mitigating gender bias by retraining on 'balanced' data

In Table 8.4 we experiment with retraining models from scratch. We carry out these experiments for the English-to-German model only since this system has the highest WinoMT accuracy and BLEU score by a large margin, meaning its counterfactual forward-translated data is likely to be higher quality than for the other language pairs. As well, it has the largest amount of data overall, meaning that the added data is likely to come from a wide spread of sources and is less likely to result in over-fitting.

System	Sentence pairs	BLEU	Acc	M:F	Δ*G*	Δ*S*
Baseline	17.2M	42.7	60.1	3.4	18.6	13.4
Downsampled	15.5M	38.2	47.9	7.1	39.8	8.0
Upsampled	18.1M	40.4	62.0	3.0	14.6	17.5
Counterfactual	18.6M	41.1	59.1	3.4	19.0	9.0

Table 8.4 General test set BLEU and WinoMT scores when training from scratch on English-German data with gendered sentence up-sampling, down-sampling and counterfactual data augmentation.

Retraining on any of the 'gender-balanced' training sets results in considerably worse general MT performance than the baseline system. The highest BLEU score is obtained by the counterfactual system, but it is still 1.6 points lower than the baseline score. The down-sampled system, in addition to a low BLEU score (4.5 points lower than the baseline), has a low accuracy score and a very high Δ G score. None of the systems show large gains over the baseline for gender accuracy.

The best gender accuracy improvement over the baseline occurs with up-sampling. Up-sampling gives a 3.2% relative improvement in accuracy, corresponding to a 5.4% relative decrease in translation quality. These results suggest that this approach to removing gender bias from MT training data prior to training without affecting translation quality is not effective. The data-centric retraining schemes not only decrease general translation performance as measured by BLEU, but also fail to significantly improve the performance of the system in relation to gender-specific metrics.

One reason for this under-performance is the presence of default inflections in gender-inflected target languages. While sentences may appear to have equal male and female entities on the English side, many apparently ungendered phrases in English would default to masculine constructions in the target language. For example, the phrase 'engineers say' would be translated to German as 'Ingenieure sagen' – a construction implying masculine gender, following German linguistic conventions. Adjusting only sentences that are identifiably gendered in English has a negligible impact on the number of default masculine constructions in the target language.

We note that all systems have very high M:F ratios, which as previously discussed reduces the relevance of ΔS. In particular, the down-sampling scheme is likely to remove rare examples of feminine constructions, resulting in a system which defaults to masculine forms for almost all German words. The result is a WinoMT M:F prediction ratio of over 7:1, a very high ΔG score (i.e., most masculine sentences correct, most feminine sentences incorrect) and a very low ΔS score (i.e., almost all entities are predicted as male, whether pro-stereotypical or anti-stereotypical).

By contrast, up-sampling does slightly improve accuracy and more significantly improves Δ G score under WinoMT, possible since the number of feminine grammatical constructions seen during training increases. This is the desired result, but male and female entities are still predicted in a ratio of 3:1, when the true test set ratio is 1:1. The up-sampling scheme also results in reduced general translation performance and an increased ΔS score. Both of these results can be attributed to over-fitting on the duplicated feminine training sentences, consolidating gender roles present in the training data. This result is, however, an indication that adding data is more likely to be effective than removing it. Training with the counterfactual dataset suffers the smallest general translation performance degradation, indicating a relative advantage of creating synthetic data.

Adaptation to semi-synthetic datasets

	en-de				**en-es**				**en-he**			
	BLEU	Acc	Δ*G*	Δ*S*	BLEU	Acc	Δ*G*	Δ*S*	BLEU	Acc	Δ*G*	Δ*S*
Baseline	42.7	60.1	18.6	13.4	27.8	49.6	36.7	2.0	**23.8**	51.3	15.1	26.4
Original	41.8	60.7	15.9	15.6	**28.3**	53.0	**24.3**	10.8	23.5	**53.6**	**12.2**	31.7
FTrans original	43.3	60.0	20.0	13.9	27.4	51.6	31.6	-4.8	23.4	48.7	23.0	**20.9**
FTrans swapped	**43.4**	63.0	15.4	12.7	27.4	**53.7**	24.5	-3.8	23.7	48.1	20.7	22.7
Balanced	42.5	**64.0**	**12.6**	**12.4**	27.7	52.8	26.2	**1.9**	**23.8**	48.3	20.8	24.0

Table 8.5 General test set BLEU and WinoMT scores after unregularized fine-tuning the baseline on four gender-based adaptation datasets. Improvements are inconsistent across language pairs.

Table 8.5 compares our baseline model with the results of unregularized fine-tuning on the counterfactual sets described in Section 8.3.1.

Fine-tuning for one epoch on *original*, a subset of the original data with gendered English stopwords, gives slight improvement in WinoMT accuracy and Δ*G* for all language pairs, while Δ*S* worsens. We suggest this set consolidates examples present in the full dataset, improving performance on gendered entities generally but emphasizing stereotypical roles.

On the *FTrans original* set Δ*G* increases sharply relative to the *original* set, while Δ*S* decreases. We suspect this set suffers from bias amplification (Zhao, Wang, et al., 2017) introduced by the baseline system during forward-translation. The model therefore over-predicts male entities even more heavily than we would expect given the gender makeup of the adaptation data's source side. Over-predicting male entities lowers Δ*S* artificially.

Adapting to *FTrans swapped* increases accuracy and decreases both Δ*G* and Δ*S* relative to the baseline for en-de and en-es. This is the desired result, but not a particularly strong one,

and it is not replicated for en-he. The *balanced* set has a very similar effect to the *FTrans swapped* set, with a smaller test BLEU difference from the baseline.

We do find that the largest improvement in WinoMT accuracy consistently corresponds to the model predicting male and female entities in the closest ratio. However, the best ratios for models adapted to these datasets are 2:1 or higher, and the accuracy improvement is small.

Overall, improvements from fine-tuning on counterfactual datasets (*FTrans swapped* and *balanced*) are present. However, they are not very different from the improvements when fine-tuning on equivalent non-counterfactual sets (*original* and *FTrans original*). Improvements are also inconsistent across language pairs.

Handcrafted profession set adaptation

		en-de				en-es				en-he			
		BLEU	Acc	Δ*G*	Δ*S*	BLEU	Acc	Δ*G*	Δ*S*	BLEU	Acc	Δ*G*	Δ*S*
1	Baseline	**42.7**	60.1	18.6	13.4	**27.8**	49.6	36.7	2.0	**23.8**	51.3	15.1	26.4
2	Balanced	42.5	64.0	12.6	12.4	27.7	52.8	26.2	**1.9**	**23.8**	48.3	20.8	24.0
3	Handcrafted (no overlap)	40.6	71.2	3.9	10.6	26.5	64.1	9.5	-10.3	23.1	56.5	-6.2	28.9
4	Handcrafted	40.8	**78.3**	**-0.7**	**6.5**	26.7	**68.6**	**5.2**	-8.7	22.9	**65.7**	**-3.3**	**20.2**

Table 8.6 General test set BLEU and WinoMT scores after fine-tuning on the handcrafted profession set, compared to fine-tuning on the most consistent counterfactual set. Lines 1-2 duplicated from Table 8.5

Results for fine-tuning on the handcrafted set are given in lines 3-4 of Table 8.6. These experiments take place in minutes on a single GPU, compared to several hours when fine-tuning on the counterfactual sets and far longer when training from scratch.

Fine-tuning on the handcrafted sets gives a much faster BLEU drop than fine-tuning on counterfactual sets. This is unsurprising since the handcrafted sets are domains of new sentences with consistent sentence length and structure, making them easy to over-fit. By contrast the counterfactual sets are less repetitive and close to subsets of the original training data, slowing forgetting.

Line 4 of Table 8.6 adapts to the handcrafted set, stopping when validation BLEU degrades by 5% on each language pair. This is approximately the same BLEU degradation as experienced when retraining with up-sampling in Table 8.4, which reached 62% WinoMT accuracy for en-de. When adapting to the handcrafted set, a similar BLEU degradation corresponds to a WinoMT accuracy up to 19 points above the baseline. This is also far more WinoMT improvement than the best counterfactual result.

When adapting to the handcrafted set difference in gender score ΔG also improves by at least a factor of 4. Stereotyping score ΔS also improves far more than for counterfactual fine-tuning. Unlike the Table 8.5 results, the improvement is consistent across all WinoMT metrics and all language pairs.

The model adapted to no-overlap handcrafted data (line 3) gives a similar drop in BLEU to the model in line 4. This model also gives stronger and more consistent WinoMT improvements over the baseline compared to the balanced counterfactual set, despite the implausibly strict scenario of no English profession vocabulary in common with the challenge set. This demonstrates that the adapted model does not simply memorize vocabulary.

8.3.4 Summary of data-centric approaches to gender bias

In this section we explore data-centric approaches to reducing gender bias in NMT. We demonstrate that, while it is possible to increase the number of feminine-inflected entities in natural bilingual data, retraining from scratch or fine-tuning on such data does not result in strong improvements in terms of gender accuracy, and general translation ability can be harmed.

Fine-tuning on a small, synthetic data-set also tends to degrade general translation performance, but allows extremely strong improvements on the gender bias test set with very little computational cost. The improvements are clear even when accounting for the possibility of vocabulary memorisation.

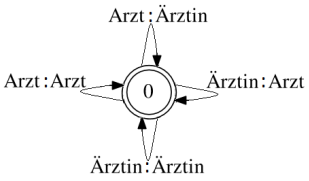
8.4 Avoiding catastrophic forgetting while adapting to reduce bias

In the previous section, we were able to significantly improve gender translation accuracy via domain adaptation to a very small synthetic dataset. However, doing so also resulted in catastrophic forgetting of general translation ability as measured by BLEU. In this section we aim to address this downside of our synthetic adaptation scheme using regularized domain adaptation and lattice-constrained inference.

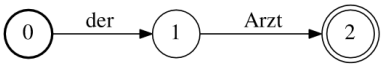
We previously explored the effectiveness of regularized training to avoid catastrophic forgetting during domain adaptation in Sec. 5.2 of this thesis. In particular, we found that EWC is generally effective at allowing improvements on the new domain while reducing forgetting. We therefore mitigate catastrophic forgetting while adapting to reduce bias by applying EWC.

Lattice constrained inference was applied in Sec. 7.3 to allow ensemble decoding with models producing different target representations. In that case the lattice defined mappings between two data representations with different levels of syntactic annotation. In this chapter we instead use lattices to define mappings between gender-inflected search spaces. The mapping is from the set of word inflections present in the original translation to a set of many alternately-gendered word inflections.

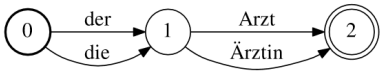
8.4.1 Rescoring gender-inflected search spaces



(a) A subset of flower transducer T. T maps vocabulary to itself as well as to differently-gendered inflections.



(b) Acceptor Y_B representing the biased first-pass translation $\mathbf{y_B}$ for source fragment 'the doctor'. The German hypothesis has the male form.



(c) Gender-inflected search space constructed from the biased hypothesis 'der Arzt'. Projection of the composition $Y_B \circ T$ contains paths with differently-gendered inflections of the original biased hypothesis. This lattice can now be rescored by an adapted model which is less affected by gender bias.

Fig. 8.2 Finite State Transducers for lattice rescoring.

Here we describe our lattice rescoring scheme for avoiding catastrophic forgetting when mitigating gender bias. We assume we have two NMT models. With one we decode fluent translations which contain gender bias (B). For the one-best hypothesis we would translate:

$$\mathbf{y_B} = \text{argmax}_{\mathbf{y}} p_B(\mathbf{y}|\mathbf{x}) \tag{8.1}$$

The other model has undergone some form of fine-tuning (FT) to reduce bias effects at a cost to translation performance, producing:

$$\mathbf{y_{FT}} = \text{argmax}_{\mathbf{y}} p_{FT}(\mathbf{y}|\mathbf{x}) \tag{8.2}$$

We construct a flower transducer T that maps each word in the target language's vocabulary to itself, as well as to other forms of the same word with different gender inflections (Figure 8.2a). We also construct Y_B, a lattice with one path representing the biased but fluent hypothesis $\mathbf{y_B}$ (Figure 8.2b).

The acceptor $\mathscr{P}(\mathbf{y_B}) = \text{proj}_{\text{output}}(Y_B \circ T)$ defines a language consisting of all the gender-inflected versions of the biased first-pass translation $\mathbf{y_B}$ that are allowed by T (Figure 8.2c). We can now decode with lattice rescoring (*LR*) by constraining inference to $\mathscr{P}(\mathbf{y_B})$:

$$\mathbf{y_{LR}} = \text{argmax}_{\mathbf{y} \in \mathscr{P}(\mathbf{y_B})} p_{FT}(\mathbf{y}|\mathbf{x}) \tag{8.3}$$

In practice we use beam search, not an argmax, to decode the various hypotheses. We construct T using heuristics on large vocabulary lists for each target language.

8.4.2 Experimental setup

We use the same models and data as in the previous section. For regularized adaptation we apply EWC when performance on the original validation set drops. We select the weighting hyperparameter Λ via validation set BLEU.

For lattice rescoring we require a transducer T containing gender-inflected forms of words in the target vocabulary. To obtain the vocabulary for German we use all unique words in the full target training dataset. For Spanish and Hebrew, which have smaller and less diverse training sets, we use 2018 OpenSubtitles word lists[4]. We then use DEMorphy (Altinok, 2018) for German, spaCy (Honnibal and Montani, 2017) for Spanish and the small set of gendered suffixes for Hebrew (Schwarzwald, 1982) to approximately lemmatize each vocabulary word and generate its alternately-gendered forms[5]. While there are almost certainly paths in T containing non-words, we expect these to have low likelihood under the rescoring models. For lattice compositions we use the efficient OpenFST implementations (Allauzen, Riley, et al., 2007).

8.4.3 Experiments in improving gender translation accuracy while maintaining translation performance

Fine-tuning to convergence and EWC regularization

The drop in BLEU and improvement on WinoMT can be explored by varying the training procedure. The model of line 5 in Table 8.7 simply adapts to handcrafted data for more

[4] Accessed Oct 2019 from https://github.com/hermitdave/FrequencyWords/.

[5] Inflection lists and scripts are available at the github https://github.com/DCSaunders/gender-debias.

		en-de				en-es				en-he			
		BLEU	Acc	ΔG	ΔS	BLEU	Acc	ΔG	ΔS	BLEU	Acc	ΔG	ΔS
1	Baseline	**42.7**	60.1	18.6	13.4	**27.8**	49.6	36.7	2.0	23.8	51.3	15.1	26.4
2	Balanced	42.5	64.0	12.6	12.4	27.7	52.8	26.2	**1.9**	23.8	48.3	20.8	24.0
3	Handcrafted (no overlap)	40.6	71.2	3.9	10.6	26.5	64.1	9.5	-10.3	23.1	56.5	-6.2	28.9
4	Handcrafted	40.8	78.3	**-0.7**	6.5	26.7	68.6	5.2	-8.7	22.9	65.7	-3.3	20.2
5	Handcrafted (converged)	36.5	**85.3**	-3.2	6.3	25.3	**72.4**	**0.8**	-3.9	22.5	**72.6**	-4.2	21.0
6	Handcrafted EWC	42.2	74.2	2.2	8.4	27.2	67.8	5.8	-8.2	23.3	65.2	**-0.4**	25.3
7	Rescore 1 with 3	**42.7**	68.3	7.6	11.8	**27.8**	62.4	11.1	-9.7	**23.9**	56.2	2.8	23.0
8	Rescore 1 with 4	**42.7**	74.5	2.1	6.5	**27.8**	64.2	9.7	-10.8	**23.9**	58.4	2.7	18.6
9	Rescore 1 with 5	42.5	81.7	-2.4	**1.5**	27.7	68.4	5.6	-8.0	23.6	63.8	0.7	**12.9**

Table 8.7 General test set BLEU and WinoMT scores after fine-tuning on the handcrafted profession set, compared to fine-tuning on the most consistent counterfactual set. Lines 1-4 duplicated from Table 8.6. Lines 5-6 vary adaptation training procedure. Lines 7-9 apply lattice rescoring to baseline hypotheses.

iterations with no regularization, to approximate loss convergence on the handcrafted set. This leads to a severe drop in BLEU, but even higher WinoMT scores.

In line 6 we regularize adaptation with EWC. There is a trade-off between general translation performance and WinoMT accuracy. With EWC regularization tuned to balance validation BLEU and WinoMT accuracy, the decrease is limited to about 0.5 BLEU on each language pair. Adapting to convergence, as in line 5, would lead to further WinoMT gains at the expense of BLEU.

The purpose of EWC regularization is to avoid catastrophic forgetting of general translation ability. This does not occur in the counterfactual experiments (e.g. line 2), with a maximum loss of 0.2 BLEU relative to the baseline, so we do not apply EWC. Moreover, WinoMT accuracy gains are small with standard fine-tuning, which allows maximum adaptation: we suspect EWC would prevent any improvements.

Lattice rescoring with less biased models

In lines 7-9 of Table 8.7 we consider lattice-rescoring the baseline output, using three models fine-tuned on the handcrafted data.

Line 7 rescores the general test set hypotheses (line 1) with a model adapted to handcrafted data that has no source language profession vocabulary overlap with the test set (line 3). This scheme shows no BLEU degradation from the baseline on any language and in fact a slight

improvement on en-he. Accuracy improvements on WinoMT are only slightly lower than for decoding with the rescoring model directly, as in line 3.

In line 8, lattice rescoring with the non-converged model adapted to handcrafted data (line 4) likewise leaves general BLEU unchanged or slightly improved. When lattice rescoring the WinoMT challenge set, 79%, 76% and 49% of the accuracy improvement is maintained on en-de, en-es and en-he respectively. This corresponds to accuracy gains of up to 30% relative to the baselines with no general translation performance loss.

In line 9, lattice-rescoring with the converged model of line 5 limits BLEU degradation to 0.2 BLEU on all languages, while maintaining 85%, 82% and 58% of the WinoMT accuracy improvement from the converged model for the three language pairs. Lattice rescoring with this model gives accuracy improvements over the baseline of 36%, 38% and 24% for en-de, en-es and en-he.

Rescoring en-he maintains a much smaller proportion of WinoMT accuracy improvement than en-de and en-es. We believe this is because the en-he baseline is particularly weak. The weakness may be due to a small and non-diverse training set, as the baseline must produce some inflection of the correct entity before lattice rescoring can have an effect on gender bias. Alternatively it may be a language-specific effect: some Hebrew gendered terms are distinguished only by vowel changes, many of which are marked only by diacritics which are usually excluded from text. The WinoMT procedure itself may therefore be noisy for English-Hebrew assessment. We note that, in general, automatic morphological analysis for Hebrew remains challenging (Tsarfaty et al., 2019).

Reducing gender bias in 'black box' commercial systems

	en-de			en-es			en-he		
	Acc	ΔG	ΔS	Acc	ΔG	ΔS	Acc	ΔG	ΔS
1	**82.0** (74.1)	-3.0 (0.0)	4.0 (30.2)	65.8 (47.3)	3.8 (36.8)	**1.9** (23.2)	63.9 (48.1)	-2.6 (14.9)	23.8 (32.9)
2	80.0 (59.4)	-3.0 (12.5)	**2.7** (12.5)	68.9 (53.1)	**0.6** (23.4)	4.6 (21.3)	**64.6** (53.7)	-1.8 (7.9)	21.5 (37.8)
3	81.8 (62.4)	**-2.6** (12.9)	4.3 (16.7)	**71.1** (59.4)	0.7 (15.4)	6.7 (22.3)	62.8 (50.5)	**-1.1** (10.3)	26.9 (47.3)
4	78.4 (48.6)	-4.0 (34.5)	5.3 (10.3)	66.0 (45.6)	4.2 (46.3)	-2.1 (15.0)	62.5 (46.6)	-2.0 (20.5)	**10.2** (24.5)

Table 8.8 We generate gender-inflected lattices from commercial system translations, collected by Stanovsky et al. (2019) (1: Microsoft, 2: Google, 3: Amazon, 4: SYSTRAN). We then rescore with the bias-reduced model from line 5 of Table 8.7. Scores are for the rescored hypotheses, with bracketed baseline scores duplicated from Table 8.3.

We note that EWC requires access to the original model parameters and training data in order to estimate the Fisher information (see Sec. 5.2.) However, lattice rescoring only requires access to the original model's translations, a gender-inflection transducer, and a

	% Non-null compositions	
	$G = T$	$G = T'$
$(Hyp_M \circ G) \circ Ref_M$	68.0	68.0
$(Hyp_M \circ G) \circ Ref_F$	57.7	68.0
$(Hyp_F \circ G) \circ Ref_M$	45.4	45.9
$(Hyp_F \circ G) \circ Ref_F$	39.7	45.9

Table 8.9 For en-de, we obtain hypothesis translations (Hyp) for the masculine (M) and feminine (F) halves of the handcrafted set using the baseline system. We compose the hypotheses with either the true gender-inflection lattice T, or an augmented version, T' containing all inflection mappings in the handcrafted reference sentences. We then compose the result with either M or F reference (Ref).

model with good gender translation accuracy to perform rescoring. The rescoring model does not necessarily need to be a fine-tuned version of the original model.

This allows us to apply lattice rescoring with the gender inflection transducer to translations of WinoMT[6] from the commercial systems listed in Table 8.3. Results are given in Table 8.8. We rescore these lattices with our model achieving the strongest WinoMT scores (line 5 of Table 8.7). We find this substantially improves WinoMT accuracy for all systems and language pairs.

One interesting observation is that WinoMT accuracy after rescoring tends to fall in a fairly narrow range for each language relative to the performance range of the baseline systems. For example, a 25.5% range in baseline en-de accuracy becomes a 3.6% range after rescoring. This suggests that our rescoring approach is not limited as much by the bias level of the baseline system as by the gender-inflection transducer and the models used in rescoring. Indeed, we emphasize that the large improvements reported in Table 8.8 do not require any knowledge of the commercial systems or the data they were trained on; we use only the translation hypotheses they produce and our own rescoring model and transducer.

Inflected search space coverage

We can investigate the limits of the gender-inflection transducer for rescoring behaviour by finding the proportion of reference sentences that exist in the rescoring search space. We conduct these experiments with the handcrafted dataset, since we have no references for WinoMT, for English-German, the language pair which reaches the highest accuracy. We use the baseline model to produce hypotheses for the masculine and feminine halves of the

[6]The raw commercial system translations are provided by the authors of Stanovsky et al. (2019) at https://github.com/gabrielStanovsky/mt_gender.

dataset. We compose these with the gender-inflected lattice, and measure the proportion of masculine and feminine reference sentences present in the resulting search spaces.

The results (Table 8.9) confirm that the baseline hypothesizes feminine forms far less frequently than masculine forms. In other words the reference is far more likely to be present in the search space if the hypothesis was masculine. Explicitly augmenting the transducer with reference mappings increases the proportion of references that are findable, suggesting weaknesses in the transducer, although reference augmentation does not compensate for the baseline system bias. We note that these results are on the handcrafted data, and therefore not necessarily a hard limit on performance for rescoring the WinoMT set.

8.4.4 Summary: mitigating catastrophic forgetting and gender bias

Simple gender-domain adaptation on a small synthetic dataset allows swift improvement in gender translation accuracy, but causes general translation ability to degrade. We demonstrate two approaches to limit this: EWC and a lattice rescoring approach. Both allow gender bias mitigation while maintaining general translation performance. Both approaches have complications: EWC requires access to the original model parameters and representative training data to compute the regularization parameters, while lattice rescoring is a two-step procedure. We find the lattice rescoring approach allows far greater improvements in gender accuracy than EWC and potentially no BLEU degradation, without requiring access to the original model or dataset.

8.5 Effects of tagged adaptation for controllable gender signals

So far in this chapter we have discussed approaches to mitigating gender bias in NMT that rely on 'gender signals'. These are typically words in the source sentence, such as gendered pronouns. An NMT system must accomplish two distinct tasks to make use of such gender signals: identifying the gender signal or feature, and then applying it to translate the relevant words in the source sentence. So far we have assumed that if we *could* correctly identify the genders of all entities in a source sentence we could translate into the target language with correct inflections, reducing the effects of gender bias.

We proceed to explore this assumption. We propose a scheme for incorporating an explicit gender inflection tag into NMT, particularly for translating coreference sentences *where the reference gender label is known*. Experimenting with translation from English to Spanish and English to German, the more successful systems from the previous section,

we find that simple existing approaches over-generalize from a gender signal, incorrectly using the same inflection for every entity in the sentence. We show that a tagged-coreference adaptation approach is effective for combatting this behaviour. Although we only work with English source sentences to extend prior work, we note that this approach can be extended to source languages without inherent gender signals like gendered pronouns, unlike approaches that rely on those signals.

Intuitively, if gender tagging does not perform well when it can use the label determined by human coreference resolution, it is likely to be less useful when a gender label must be automatically inferred. Conversely, gender tagging that is effective in this scenario may be beneficial when the user can specify the gender of the referent, such as Google Translate's translation inflection selection (Johnson, 2018), or for translations where the genders of all human referents are known. We also explore automatic gender tagging for English test sentences for cases in which the genders of human referents are not known.

Existing work in NMT gender bias has focused on the translation of sentences based on binary gender signals, such as exclusively male or female personal pronouns. This effectively highlights gender biases in translation between masculine and feminine referents. However, it also excludes and erases those who do not use binary gendered language, including but not limited to non-binary individuals (Cao and Daumé III, 2020; Zimman, 2017). Using synthetic adaptation data with gender tags allows us to define new, controllable gender inflection translations. We therefore explore applying tagging to indicate gender-neutral referents, and produce a WinoMT-style test set to assess translation of coreference sentences with gender-neutral entities.

8.5.1 Assessing second-entity and neutral translation

Feature over-generalization and second-entity translation

We note a comment by Rudinger et al. (2018), who develop a portion of the English WinoMT source sentences, that Winograd schemas 'may demonstrate the presence of gender bias in a system, but not prove its absence.' The authors review manifestations of gender bias in language which are not analysed at all by the schemas. However we here consider one example, gender-feature over-generalization, which can be further assessed with WinoMT. Considering the previous WinoMT example:

The developer argued with the designer because she did not like the design.

For this sentence, high WinoMT accuracy can be achieved by using the labelled gender inflection, or equivalently the inflection of the gendered pronoun, for both primary and

Label	Original WinoMT			Our WinoMT		
	Full	Pro	Anti	Secondary	Neutral	Neutral secondary
Male	1826	792	794	1826	0	0
Female	1822	792	790	1822	0	0
Neutral	240	0	0	240	1826	1826

Table 8.10 Summary of sentence counts for different gender labels for WinoMT. Original WinoMT sets are from Stanovsky et al. (2019) (full, pro-sterotypical and anti-stereotypical). Our extended WinoMT sets assess secondary entities in original WinoMT, neutral-labelled primary entities, and secondary entities in the neutral-labelled primary entities set.

secondary entities. This is true for all WinoMT test sentences, even though each sentence only specifies the gender of the primary entity.

We therefore produce a test set for the WinoMT framework to track the gender inflection of the secondary entity in each original WinoMT sentence (e.g. 'the designer' in the above example). We measure second-entity inflection correspondence with the gender label, which we refer to as **L2**. High L2 suggests that 'the designer' would also have feminine inflection in a translation of the above example, despite not being coreferent with the pronoun.

We are particularly interested in cases where L2 increases over a baseline, or high **ΔL2**. Many factors may contribute to a baseline system's L2, but we are specifically interested in whether *adding* gender features affects only the words they are intended to affect. High ΔL2 indicates a system learning to over-generalize from available gender features. We consider this as erasing the secondary referents, and therefore as undesirable behaviour.

Exploring gender-neutral translation

We wish to extend previous machine translation coreference research to the translation of gender-neutral language, which may be used by non-binary individuals or to avoid the social impact of using gendered language (Misersky et al., 2019; Zimman, 2017). Recently Cao and Daumé III (2020) have encouraged inclusion of non-binary referents in NLP coreference work. Their study focuses heavily on English, where gender-neutral language such as singular *they* is in increasingly common use (Bradley et al., 2019); the authors acknowledge that 'some extensions ... to languages with grammatical gender are non-trivial'.

In particular, existing NMT gender bias test sets typically analyse behaviour in languages with grammatical gender that corresponds to a referent's gender. Translation into these languages is effective in highlighting differences in translation between masculine and feminine referents, but these languages also often lack widely-accepted conventions for gender-neutral language (Ackerman, 2019; Hord, 2016). In some languages with binary

grammatical gender it is possible to avoid gendering referents by using passive or reflexive grammar, but such constructions can themselves invalidate individual identities (Auxland, 2020).

We therefore explore a proof-of-concept scheme for translating tagged neutral language into inflected languages by introducing synthetic gender-neutral placeholder articles and noun inflections in the target language. For example, we represent the gender-neutral inflection of 'el entrenador' (the trainer) as 'DEF entrenadorW_END'

A variety of gender-neutral inflections have been proposed for various grammatically gendered languages, such as *e* or *x* Spanish (Papadopoulos, 2019) and Portuguese (Auxland, 2020) noun inflections instead of masculine *o* and feminine *a*. These language-specific approaches may develop in various forms within and across social networks, and can shift over time (Shroy, 2016). Our intent is not to prescribe which should be used, but to explore an approach which in principle could be extended to various real inflection schemes.

We construct additional 'neutral-augmented' versions of the adaptation sets described in 8.5.4, adding '*The [adjective] person finished [their|the] work*' sentences to the adjective-based sets and sentences like '*The trainer [N] finished [their|the] work*' to the profession-based sets, with synthetic placeholder articles DEF and inflections W_END on the target side of profession sentences. We give examples for Spanish and German in Table 8.11. We also construct a neutral-label-only version of WinoMT containing the 1826 unique binary templates filled with they/them/their. We report results on the original and neutral-augmented sets separately for ease of comparison with prior work.

8.5.2 Controlling gender inflection

We wish to investigate whether a system can translate into inflected languages correctly correctly given the reference gender label of a certain word. Our proposed approach involves fine-tuning a model on a synthetic set of sentences which have gender tags. At test time we assign the reference gender label to the words whose gender inflection we wish to control. In the example of the first WinoMT sentence discussed earlier, the gender label is 'female' and the primary entity is 'the developer', so the tagged sentence becomes:

The developer [F] argued with the designer because she did not like the design.

We only tag the primary entity in test sentences, but also assess the inflection of the secondary entity in response to these tags using the secondary-entity set discussed above.

As our baseline, we take the handcrafted no-overlap set described in Sec. 8.3.1, as it allows strong improvements in WinoMT accuracy while avoiding the confounding effects of vocabulary memorization. In this section we refer to this set as **V0** for brevity.

	English source	German target	Spanish target
V0	the trainer finished his work the trainer finished her work the trainer finished their work	der Trainer beendete seine Arbeit die Trainerin beendete ihre Arbeit DEF TrainerW_END beendete PRP Arbeit	el entrenador terminó su trabajo la entrenadora terminó su trabajo DEF entrenadorW_END terminó su trabajo
V1	the trainer [M] finished his work	der Trainer beendete seine Arbeit	el entrenador terminó su trabajo
V2	the trainer [F] finished the work	die Trainerin beendete die Arbeit	la entrenadora terminó el trabajo
V3	the trainer [N] and the choreographer [M] finished the work	DEF TrainerW_END und der Choreograf beendeten die Arbeit	DEF entrenadorW_END y el coreógrafo terminaron el trabajo
V4	the trainer [F], the choreographer [N]	die Trainerin, DEF ChoreografW_END	la entrenadora, DEF coreógrafW_END

Table 8.11 Examples of the tagging schemes explored in this chapter. Adjective-based sentences (e.g. 'the tall woman finished her work') are never tagged. For neutral target sentences, we define synthetic placeholder articles DEF and noun inflections W_END, as well as a placeholder possessive pronoun for German PRP

We then propose four gender-tagged variations on V0 which we illustrate in Table 8.11. In the first, **V1**, we add a gender tag following professions only (we do not tag adjective-based sentences since 'man' and 'woman' are already distinct words in English).

For the second, **V2**, we use the same tagging scheme but note that the possessive pronoun offers a gender signal that may conflate with the tag, so change all examples to '... finished *the* work'.

The third, **V3**, is the same as **V2** but in each profession-based sentence a second profession-based entity with a different gender inflection tag is added. This set is intended to discourage systems from over-generalizing one tag to all sentence entities.

In the final scheme, **V4**[7], we simplify **V3** to a minimal, lexicon-like pattern:

The [entity1], the [entity2].

Both entities are tagged. We remove all adjective-based sentences, leaving only tagged coreference entities for adaptation. This set has the advantage of using simpler language than other sets, making it easier to extend to new target languages.

8.5.3 Experimental setup

For this task, we assess English-to-German and English-to-Spanish NMT, as these were the systems that saw strong gender accuracy improvement under previous approaches. Models and baseline training are as described in the previous sections. We define gender tags as

[7] V4 proposed by R. Sallis as part of a MEng thesis in progress (Sallis, 2021).

System	Labelled WinoMT	en-de			en-es		
		BLEU	Acc	ΔL2	BLEU	Acc	ΔL2
Baseline	×	42.7	60.1	-	27.8	49.6	-
V0	×	42.4	82.3	27.4	27.7	66.3	29.7
V1	✓	42.5	81.7	26.6	27.7	69.0	26.4
V2	✓	42.5	**84.1**	24.2	27.5	70.9	13.2
V3	✓	42.6	77.4	**1.1**	27.5	80.6	**0.3**
V4	✓	42.6	80.6	2.0	27.6	**83.1**	8.7

Table 8.12 Test BLEU, WinoMT primary-entity accuracy (Acc), and change in second-entity label correspondence ΔL2. We adapt the baseline to a synthetic set without tags (V0), or to one of the binary gender-inflection tagging schemes (V1-4). 'Labelled WinoMT' indicates whether WinoMT primary entities are tagged with their reference gender label. All results are for rescoring the baseline system gendered-alternative lattices with the listed model.

unique vocabulary items which only appear in the source sentence. We adapt to synthetic data with minibatches of 256 tokens for 64 training updates, which we found gave the strongest accuracy improvements when fine-tuning on the handcrafted no-overlap (V0) datasets. This results in different BLEU score and WinoMT accuracy from the results of Table 8.7.

The V3 sets have about 30% more tokens, the V4 sets about 30% fewer and the neutral-augmented sets about 50% more: we adjust the adaptation steps accordingly for these cases.[8]

For all results we rescore the baseline system gendered-alternative lattices with the listed model as described in Sec. 8.4.1. This constrains the output hypothesis to be a gender-inflected version of the original baseline hypothesis. For the gender-neutral experiments we add synthetic inflections and articles to the lattices.

When assessing automatic test set tagging we use the RoBERTa (Liu, Ott, et al., 2019) pronoun disambiguation function tuned on Winograd Schema Challenge data as described in Fairseq documentation[9].

8.5.4 Avoiding over-generalization with tagging schemes

Measured improvements in gender accuracy are often accompanied by over-generalization

Table 8.12 gives BLEU score and primary-entity accuracy for the original, binary versions of synthetic adaptation sets described in section 8.5.4. WinoMT test sentences have primary entities tagged with their gender label if the adaptation set had tags, and are unlabelled

[8]Adaptation and tagged test sets are available at the github https://github.com/DCSaunders/tagged-gender-coref.

[9]https://github.com/pytorch/fairseq/tree/master/examples/roberta/wsc.

otherwise. We note that lattice rescoring keeps the general test set score within 0.3 BLEU of the baseline, and focus on the variation in WinoMT performance.

Primary-entity WinoMT accuracy does increase significantly over the baseline for all adaptation schemes. V3 and V4, which contain multi-entity examples, are particularly effective for en-es, while V2, which contains a single entity, is more effective for en-de. We hypothesize this reflects the difference in baseline quality: the stronger en-de baseline is more likely to have already seen multiple-entity sentences.

We also report ΔL2, the change in the secondary entity's label correspondence compared to the baseline. High ΔL2 implies that the model is over-generalizing a gender signal intended for the primary entity to the secondary entity. In other words, the gender signal intended for the primary entity has a very strong influence on the translation of the secondary entity. ΔL2 does indeed increase strongly from the baseline for the V0 and V1 systems, confirming our suspicion that these models trained on sentences with a single entity simply learn to apply any gender feature to both entities in the test sentences indiscriminately.

Remarkably, for adaptation to V0 and V1 datasets we found that the secondary entity is inflected to correspond with the pronoun more often than the primary entity which is labelled as coreferent with it. A possible explanation is that the secondary entity occurs at the start of the sentence in about two thirds of test sentences, compared to about one third for the primary entity. Adapting to single-entity test sets may encourage the model to simply inflect the first entity in the sentence using the gender signal.

For V2, where the source possessive pronoun is removed and the tag is the only gender signal, ΔL2 still increases significantly, although less than for V1. This indicates that even if the only signal is a gender tag applied directly to the correct word, it may be wrongly taken as a signal to inflect other words. The V3 scheme is the most promising, with a 17% increase in accuracy for en-de and a 30% increase for en-es corresponding to very small changes in L2, suggesting this model minimizes over-generalization from gender features beyond the tagged word. V4 performs similarly to V3 for en-de but suffers from an L2 increase for en-es. It is possible that a lexicon-style set with tags in every example may cause undesirable over-generalisation.

Labelled and unlabelled test sentences

Table 8.13 lists accuracy and ΔL2 with and without WinoMT source sentence labelling for the same systems as Table 8.12. V1 gives similar performance to the original V0 set with and without WinoMT labelling. Removing the possessive pronoun as in V2 decreases accuracy compared to V1 without labelling and slightly increases it with labelling, suggesting removing the source pronoun forces the model to rely on the gender tag.

System	en-de						en-es					
	Unlabelled		Auto-labelled		Ref labelled		Unlabelled		Auto-labelled		Ref labelled	
	Acc	ΔL2	Acc	ΔL2	Acc	ΔL2	Acc	ΔL2	Acc	ΔL2	Acc	ΔL2
Baseline	60.1	-	-	-	-	-	49.6	-	-	-	-	-
V0	**82.3**	27.4	-	-	-	-	66.3	29.7	-	-	-	-
V1	81.5	26.6	81.7	26.5	81.7	26.6	**67.3**	29.6	68.5	31.2	69.0	26.4
V2	71.2	9.2	**83.6**	24.8	**84.1**	24.2	52.1	3.5	69.7	18.4	70.9	13.2
V3	57.5	-5.8	79.9	**3.7**	77.4	**1.1**	47.9	-2.5	77.7	6.4	80.6	**0.3**
V4	60.5	**-2.0**	79.2	4.6	80.6	2.0	48.5	**-0.6**	**80.6**	12.6	**83.1**	8.7

Table 8.13 WinoMT accuracy and change in second-entity label correspondence for the adaptation schemes in Table 8.12 when changing how tags are determined for **WinoMT source sentences**. The primary entity's gender label in each test sentence is either unlabelled, auto-labelled with RoBERTa, or labelled with the reference gender.

Accuracy under V2-4 improves dramatically when gender labels are added to WinoMT primary entities. Without labels the accuracy for these systems improves far less or not at all. This is unsurprising: the gender tag is the only way to infer the correct target inflection when adapting to these datasets. Nevertheless some accuracy improvement is still possible with neither tags nor possessive pronouns, possibly because the model 'sees' more examples of profession constructions in the target language.

Without test set labels, the V3 and V4 systems have negative ΔL2, implying that the second entity's inflection corresponds to the primary entity label less often than for the baseline. This is not necessarily bad, as they are still low absolute values. Small absolute ΔL2 indicates that added primary-entity gender signals have little impact on the secondary entity relative to the baseline, which is the desired behaviour. Small negative values are therefore better than large positive values.

Auto-labelling WinoMT source sentences using RoBERTa gives only slightly poorer results compared to using reference labels[10]. We find that the automatic tags agree with human tags for 84% of WinoMT sentences, with no difference in performance between masculine- and feminine-labelled sentences, or pro- and anti-stereotypical sentences. This is encouraging, and suggests that the tagged inflection approach may also be applicable to natural text, for which manual labelling is often impractical.

Gender-neutral translation

In Table 8.14 we report on systems adapted to the neutral-augmented synthetic sets, evaluated on the neutral-only WinoMT set. We use test labelling for all cases where models are trained

[10]Investigations into options for automatically labelling the WinoMT test sets were carried out by R. Sallis for a MEng thesis in progress (Sallis, 2021).

System	Labelled WinoMT	en-de		en-es	
		Acc	ΔL2	Acc	ΔL2
Baseline	×	2.7	-	4.2	-
V0	×	13.5	28.8	6.4	3.9
V1	✓	**27.3**	28.2	25.4	25.1
V2	✓	23.0	39.6	32.1	27.5
V3	✓	20.2	18.7	38.8	10.0
V4	✓	19.4	**4.4**	**56.5**	**0.7**

Table 8.14 Primary-entity accuracy and second-entity label correspondence ΔL2 on a neutral-label-only extension of WinoMT. Here, adaptation sets and lattices are augmented with synthetic neutral articles and nouns. 'Labelled WinoMT' indicates whether each sentence is tagged with its reference (neutral) gender label.

with tags. As with the binary experiments we found that performance was poor when test sentences were untagged.

Unsurprisingly, the baseline model is unable to generate the newly defined gender-neutral articles or noun inflections – the non-zero accuracy results from existing WinoMT sentences with neutral entities like 'someone'. Adapting on the neutral-augmented V0 set does little better for en-es, although it gives a larger gain for en-de. This discrepancy may be because the only neutral gender signal in the V0 source sentences is from the possessive pronoun *their*. In Spanish, which has one gender-neutral third-person singular pronoun, 'their' has the same Spanish translation as *his* or *her* and therefore does not constitute a strong gender signal, while in German we add a synthetic singular gender-neutral pronoun, which indicates neutral gender even without tags.

Adding a gender tag significantly improves primary entity accuracy. As with Table 8.12, there is little difference in labelled-WinoMT performance when the possessive pronoun is removed. Also as previously, the V3 and V4 'tagged coreference' sets shows far less over-generalization in terms of ΔL2 than the other tagged schemes, although V4 significantly outperforms V3 for en-es on this set.

We note that primary-entity accuracy is relatively low compared to results for the original WinoMT set, with our best-performing system reaching 56.5% accuracy. We consider this unsurprising since the model has never encountered most of the neutral-inflected occupation terms before, even during adaptation, due to the lack of overlap between the adaptation and WinoMT test sets. However, it does suggest that more work remains for introducing novel gender inflections for NMT.

8.5.5 Summary of tagged adaptation for controllable gender signals

Tagging words with target language gender inflection is a powerful way to improve accuracy of translated inflections. Tagging could be applied in cases where the gender of a referent is known, or as monolingual coreference resolution tools improve sufficiently to be used for automatic tagging. The scheme also has potential application to new inflections defined for gender-neutral language.

However, there is a risk that gender features will be used in an over-general way. Providing a strong gender signal for one entity can cause harm by erasing other entities in the same sentence, unless a model is specifically trained to translate sentences with multiple entities. In particular we find that systems trained on multiple-entity translation examples allow good performance while minimizing peripheral effects.

8.6 Conclusions

We treat the presence of gender bias in NMT systems as a domain adaptation problem. We explore various data-centric approaches to adjusting demonstrate strong improvements under the WinoMT challenge set by adapting to tiny, synthetic datasets with equal numbers of masculine and feminine entities for three language pairs. We show that this approach can be further extended to translation of gender-neutral entities.

We also explore regularized adaptation and lattice rescoring techniques to limit degradation in general translation ability, in the latter case without requiring access to the original model or data. In general our schemes involve encouraging the model to rely on 'gender signals', whether gendered terms or explicit tags, rather than relying heavily on demographics represented in training data.

We also investigate some previously unexplored side-effects of such approaches, such as over-generalization of a gender feature. We emphasize that work on gender coreference in translation requires care to ensure that the effects of interventions are as intended, as well as testing scenarios that capture the full complexity of the problem, if the work is to have an impact on gender bias.

Overall, we find that small-domain adaptation has great potential as more effective and efficient approach to reducing bias effects in machine translation than counterfactual data augmentation. We do not claim to fix the bias problem in NMT, but demonstrate that the effects of gender bias can be reduced without degradation in overall translation quality.

Chapter 9

Conclusions

This thesis aimed to explore domain adaptation for NMT with an eye towards the possibility of multi-domain scenarios. The potential risks of neglecting unknown-domain or multi-domain translation scenarios are particularly relevant as industrial NMT providers increasingly compete over the ability to adapt to small quantities of customer data (Savenkov, 2018). Often these approaches neglect the attendant risks of catastrophic forgetting, domain overfitting and mismatch, and exposure bias effects. While architectural changes or retraining may avoid these complications for a given domain with sufficient training data, such methods are potentially expensive and rely heavily on foreknowledge of the test domain. Instead, we have explored simple adjustments to the parameter adaptation and inference procedure to improve domain adaptation for NMT.

At the start of this thesis we posed five research questions concerning domain adaptation for NMT. In this concluding chapter we discuss how this thesis has addressed those questions and review our contributions.

9.1 How effective are data-centric approaches to NMT domain adaptation?

Simple domain-specific data selection, followed by straightforward continued training of the model on the new data, can be seen as a data-centric approach to domain adaptation. We explore this approach in Chapter 4 of this thesis, as well as treating it as a baseline approach in Chapter 5.

In both chapters we do indeed find that selecting domain-relevant training data and performing unregularized MLE fine-tuning achieves extremely strong results on individual domains. In particular we achieve state-of-the-art results for biomedical translation shared

tasks. However, these findings come with two important caveats. Firstly, we find that data-centric approaches to adaptation can lead to side-effects of exposure bias and catastrophic forgetting of other domains. Secondly, we find that data-centric approaches were outperformed by methods which also changed adaptation or inference procedure. This was partially due to the same side-effects of domain adaptation. As well, however, results in Chapter 5 suggest that robust tuning techniques like EWC and doc-MRT may simply give better model convergence.

9.2 Given an adaptation dataset, what training schemes might improve machine translation quality?

As we found in answering RQ1, data-centric fine-tuning can lead to side-effects of exposure bias and catastrophic forgetting. Our most successful schemes for combatting these side-effects involved approaches which were not simply data-centric, such as changes to the adaptation algorithm in Chapter 5 (EWC regularization and document Minimum Risk Training) or use of adaptive multi-domain ensemble weighting at inference time in Chapter 6. At inference time, we showed that adaptive ensemble weighting schemes could outperform the 'oracle' model trained or fine-tuned with data from a single domain.

9.3 Can domain adaptation help when the test domain is unknown?

Most domain adaptation approaches assume that the test domain is known and fixed. Some also assume that we have training and validation data that matches the test domain, generally by using provenance as a surrogate for domain. In this thesis we also use provenance 'domain' labels for ease of reporting and comparison, but otherwise attempt to relax these assumptions.

In Chapter 5 we show that adapting sequentially across domains with regularization can achieve good cross-domain performance. In Chapter 6 we show that unknown-domain adaptive ensembling can out-perform approaches using oracle information – that is, choosing the model domain based on known test data provenance. We show that adaptive inference is complementary to multi-domain models, as this is the case even when the oracle models have been sequentially adapted for strong performance across multiple domains.

9.4 Can changing data representation have similar effects to changing data domain?

Unlike data domain, data representation does not change text content or correspond to text provenance. However, in Chapter 7, we demonstrate that combining NMT models which use different data representations can benefit translation quality. We therefore combine multiple data representations either in a single model or in an ensemble in a way reminiscent of multi-domain translation. In particular, we develop a scheme for ensembles of models producing multiple target language representations, and show that multi-representation ensembles improve syntax-based NMT.

9.5 Can gender bias in NMT systems be mitigated by treating it as a domain?

In Chapter 8 we show gender bias effects are strongly influenced by vocabulary distributions in the training data, a key hallmark of a domain. We also show that data selection methods, particularly tuning on a synthetic dataset with carefully-selected gender features, have a strong effect on apparent model gender bias. We apply techniques developed elsewhere in the thesis to the problem of tuning on this set, specifically regularized adaptation and multi-domain inference. We show that gender bias effects in machine translation can be treated as a domain, and that the effects can be mitigated without impact on general translation quality.

9.6 Final remarks

A human can integrate knowledge from multiple new sources across the course of their life, and use it to inform their decisions, actions and indeed language. NMT systems do not currently behave in this way as a matter of course. However, this thesis has explored a number of possible avenues towards such behaviour.

With our work on the translation of gendered language in particular, we highlighted that human language is both complex and evolving, as are the contexts in which we interact with NLP tools. With this thesis we hope to draw attention to the possible benefits and drawbacks of different approaches to domain adaptive machine translation, as well as their possible applications. We hope that future work on adaptive NMT will focus not only on the language of immediate interest but the machine translation abilities or tendencies that we wish to maintain or abandon.

References

Abbasi, M., Friedler, S. A., Scheidegger, C., & Venkatasubramanian, S. (2019). Fairness in representation: Quantifying stereotyping as a representational harm. *Proceedings of the 2019 SIAM International Conference on Data Mining*, 801–809.

Ackerman, L. (2019). Syntactic and cognitive issues in investigating gendered coreference. *Glossa: a journal of general linguistics*, *4*(1).

Agrawal, S., & Carpuat, M. (2019). Controlling text complexity in neural machine translation. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1549–1564.

Aharoni, R., & Goldberg, Y. (2017). Towards string-to-tree neural machine translation. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, *2*, 132–140.

Aharoni, R., & Goldberg, Y. (2020). Unsupervised domain clusters in pretrained language models. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7747–7763.

Aljundi, R., Lin, M., Goujaud, B., & Bengio, Y. (2019). Gradient based sample selection for online continual learning. *Advances in Neural Information Processing Systems*, 11816–11825.

Allauzen, C., & Riley, M. (2011). Bayesian Language Model Interpolation for Mobile Speech Input. *Proceedings of the Twelfth Annual Conference of the International Speech Communication Association*.

Allauzen, C., Riley, M., Schalkwyk, J., Skut, W., & Mohri, M. (2007). Openfst: A general and efficient weighted finite-state transducer library. *International Conference on Implementation and Application of Automata*, 11–23.

Altinok, D. (2018). DEMorphy, German language morphological analyzer. *arXiv preprint arXiv:1803.00902*.

Alvarez-Melis, D., & Jaakkola, T. (2017). A causal framework for explaining the predictions of black-box sequence-to-sequence models. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 412–421.

Arthur, P., Neubig, G., & Nakamura, S. (2016). Incorporating discrete translation lexicons into neural machine translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1557–1567.

Assylbekov, Z., Takhanov, R., Myrzakhmetov, B., & Washington, J. N. (2017). Syllable-aware neural language models: A failure to beat character-aware ones. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1866–1872.

Ataman, D., Negri, M., Turchi, M., & Federico, M. (2017). Linguistically motivated vocabulary reduction for neural machine translation from turkish to english. *The Prague Bulletin of Mathematical Linguistics*, *108*(1), 331–342.

Auxland, M. (2020). Para Todes: A Case Study on Portuguese and Gender-Neutrality. *Journal of Languages, Texts and Society*, *4*, 1–23.

Axelrod, A. (2017). Cynical selection of language model training data. *arXiv preprint arXiv:1709.02279*.

Axelrod, A., He, X., & Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 355–362.

Ayana, S. S., Liu, Z., & Sun, M. (2016). Neural headline generation with minimum risk training. *arXiv preprint arXiv:1604.01904*.

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

Bahdanau, D., Brakel, P., Xu, K., Goyal, A., Lowe, R., Pineau, J., Courville, A., & Bengio, Y. (2017). An actor-critic algorithm for sequence prediction. *5th International Conference on Learning Representations, ICLR*.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *ICLR*.

Bañón, M., Chen, P., Haddow, B., Heafield, K., Hoang, H., Esplà-Gomis, M., Forcada, M. L., Kamran, A., Kirefu, F., Koehn, P., Ortiz Rojas, S., Pla Sempere, L., Ramírez-Sánchez, G., Sarrías, E., Strelec, M., Thompson, B., Waites, W., Wiggins, D., & Zaragoza, J. (2020). ParaCrawl: Web-scale acquisition of parallel corpora. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4555–4567.

Bapna, A., & Firat, O. (2019). Simple, scalable adaptation for neural machine translation. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1538–1548.

Barone, A. V. M., Haddow, B., Germann, U., & Sennrich, R. (2017). Regularization techniques for fine-tuning in Neural Machine Translation. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1489–1494.

Barrault, L., Bojar, O., Costa-jussà, M. R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Koehn, P., Malmasi, S., Monz, C., Müller, M., Pal, S., Post, M., & Zampieri, M. (2019). Findings of the 2019 conference on machine translation (WMT19). *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 1–61.

Basta, C., Costa-jussà, M. R., & Fonollosa, J. A. R. (2020). Towards mitigating gender bias in a decoder-based neural machine translation model by adding contextual information. *Proceedings of the The Fourth Widening Natural Language Processing Workshop*, 99–102.

Baum, E. B., & Wilczek, F. (1988). Supervised learning of probability distributions by neural networks. *Neural information processing systems*, 52–61.

Bawden, R., Bretonnel Cohen, K., Grozea, C., Jimeno Yepes, A., Kittner, M., Krallinger, M., Mah, N., Neveol, A., Neves, M., Soares, F., Siu, A., Verspoor, K., & Vicente Navarro, M. (2019). Findings of the WMT 2019 biomedical translation shared task: Evaluation for MEDLINE abstracts and biomedical terminologies. *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, 29–53.

Bawden, R., Di Nunzio, G. M., Grozea, C., Unanue, I. J., Yepes, A. J., Mah, N., Martinez, D., Névéol, A., Neves, M., Oronoz, M., Perez De Viñaspre, O., Piccardi, M., Roller, R.,

Siu, A., Thomas, P., Vezzani, F., Navarro, M. V., Wiemann, D., & Yeganova, L. (2020). Findings of the WMT 2020 Biomedical Translation Shared Task: Basque, Italian and Russian as New Additional Languages. *5th Conference on Machine Translation*.

Bender, E. M., & Friedman, B. (2018). Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, *6*, 587–604.

Bengio, S., Vinyals, O., Jaitly, N., & Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in Neural Information Processing Systems*, 1171–1179.

Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. *Journal of machine learning research*, *3*(Feb), 1137–1155.

Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. *Proceedings of the 26th annual international conference on machine learning*, 41–48.

Bengioy, Y., Simardy, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw*, *5*, 157–166.

Berard, A., Calapodescu, I., & Roux, C. (2019). Naver labs Europe's systems for the WMT19 machine translation robustness task. *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 526–532.

Bishop, C. M. (1995). Training with noise is equivalent to tikhonov regularization. *Neural computation*, *7*(1), 108–116.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, *3*(Jan), 993–1022.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Jimeno Yepes, A., Koehn, P., Logacheva, V., Monz, C., Negri, M., Névéol, A., Neves, M., Popel, M., Post, M., Rubino, R., Scarton, C., Specia, L., Turchi, M., … Zampieri, M. (2016). Findings of the 2016 conference on machine translation. *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, 131–198.

Bojar, O., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Koehn, P., & Monz, C. (2018). Findings of the 2018 conference on machine translation (WMT18). *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 272–303.

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., & Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. *Advances in neural information processing systems*, 4349–4357.

Bradley, E. D., Salkind, J., Moore, A., & Teitsort, S. (2019). Singular 'they' and novel pronouns: Gender-neutral, nonbinary, or both? *Proceedings of the Linguistic Society of America*, *4*(1), 36–1.

Britz, D., Goldie, A., Luong, M.-T., & Le, Q. (2017). Massive exploration of neural machine translation architectures. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1442–1451.

Britz, D., Le, Q., & Pryzant, R. (2017). Effective domain mixing for neural machine translation. *Proceedings of the Second Conference on Machine Translation*, 118–126.

Buciluǎ, C., Caruana, R., & Niculescu-Mizil, A. (2006). Model compression. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 535–541.

Cao, Y. T., & Daumé III, H. (2020). Toward gender-inclusive coreference resolution. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4568–4595.

Cettolo, M., Niehues, J., Stüker, S., Bentivogli, L., Cattoni, R., & Federico, M. (2016). The IWSLT 2016 evaluation campaign. *IWSLT 2016, International Workshop on Spoken Language Translation*.

Cettolo, M., Niehues, J., Stüker, S., Bentivogli, L., & Federico, M. (2014). Report on the 11th IWSLT evaluation campaign, IWSLT 2014. *Proceedings of the International Workshop on Spoken Language Translation*.

Chen, B., Cherry, C., Foster, G., & Larkin, S. (2017). Cost weighting for neural machine translation domain adaptation. *Proceedings of the First Workshop on Neural Machine Translation*, 40–46.

Chen, M. X., Firat, O., Bapna, A., Johnson, M., Macherey, W., Foster, G., Jones, L., Schuster, M., Shazeer, N., Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Chen, Z., Wu, Y., & Hughes, M. (2018). The best of both worlds: Combining recent advances in neural machine translation. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 76–86.

Cherry, C., Foster, G., Bapna, A., Firat, O., & Macherey, W. (2018). Revisiting character-based neural machine translation with capacity and compression. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4295–4305.

Chinea-Ríos, M., Peris, Á., & Casacuberta, F. (2017). Adapting neural machine translation with parallel synthetic data. *Proceedings of the Second Conference on Machine Translation*, 138–147.

Cho, K., van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder–decoder approaches. *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, 103–111.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1724–1734.

Chu, C., Dabre, R., & Kurohashi, S. (2017). An empirical comparison of domain adaptation methods for neural machine translation. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 385–391.

Chung, J., Cho, K., & Bengio, Y. (2016). A character-level decoder without explicit segmentation for neural machine translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1693–1703.

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. *Proceedings of the 25th International Conference on Machine Learning*, 160–167.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, *12*(null), 2493–2537.

Consortium, U. (2000). *The unicode standard, version 3.0* (Vol. 1). Addison-Wesley Professional.

Corbett, G. G., Fraser, N. M., & Unterbeck, B. (1999). Default genders. *Gender in Grammar and Cognition: I. Approaches to Gender, II. Manifestations of Gender*, 55–97.

Costa-jussà, M. R., Escolano, C., & Fonollosa, J. A. R. (2017). Byte-based neural machine translation. *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, 154–158.

Costa-jussà, M. R., & Fonollosa, J. A. R. (2016). Character-based neural machine translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 357–361.

Crawford, K. (2017). The trouble with bias. *Conference on Neural Information Processing Systems, invited speaker*.

Currey, A., & Heafield, K. (2019). Incorporating source syntax into transformer-based neural machine translation. *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, 24–33.

Currey, A., Mathur, P., & Dinu, G. (2020). Distilling multiple domains for neural machine translation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4500–4511.

Dabre, R., Kunchukuttan, A., Fujita, A., & Sumita, E. (2018). NICT's participation in WAT 2018: Approaches using multilingualism and recurrently stacked layers. *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation: 5th Workshop on Asian Translation: 5th Workshop on Asian Translation*.

Dahlmeier, D., & Ng, H. T. (2012). Better evaluation for grammatical error correction. *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 568–572.

Dahlmeier, D., Ng, H. T., & Wu, S. M. (2013). Building a large annotated corpus of learner English: The NUS corpus of learner English. *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, 22–31.

Dakwale, P., & Monz, C. (2017). Fine-tuning for Neural Machine Translation with limited degradation across in-and out-of-domain data. *Proceedings of the 16th Machine Translation Summit (MT-Summit 2017)*, 156–169.

Darwish, I., & Sayaheen, B. (2019). Manipulating titles in translation. *Journal of Educational and Social Research*, *9*(3), 239–239.

Daumé III, H., & Jagarlamudi, J. (2011). Domain adaptation for machine translation by mining unseen words. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 407–412.

de Gispert, A., Iglesias, G., Blackwood, G., Banga, E. R., & Byrne, W. (2010). Hierarchical phrase-based translation with weighted finite-state transducers and shallow-n grammars. *Computational Linguistics*, *36*(3), 505–533.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186.

Dietterich, T. G. (2000). Ensemble methods in machine learning. *International workshop on multiple classifier systems*, 1–15.

Ding, S., Renduchintala, A., & Duh, K. (2019). A call for prudent choice of subword merge operations in neural machine translation. *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*, 204–213.

Dinu, G., Mathur, P., Federico, M., & Al-Onaizan, Y. (2019). Training neural machine translation to apply terminology constraints. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3063–3068.

Domingo, M., Garcıa-Martınez, M., Helle, A., Casacuberta, F., & Herranz, M. (2018). How much does tokenization affect neural machine translation? *arXiv preprint arXiv:1812.08621*.

Dou, Z.-Y., Anastasopoulos, A., & Neubig, G. (2020). Dynamic data selection and weighting for iterative back-translation, 5894–5904.

Duan, C., Chen, K., Wang, R., Utiyama, M., Sumita, E., Zhu, C., & Zhao, T. (2020). Modeling future cost for neural machine translation. *arXiv preprint arXiv:2002.12558*.

Dušek, O., Hajič, J., Hlaváčová, J., Libovický, J., Pecina, P., Tamchyna, A., & Urešová, Z. (2017). Khresmoi summary translation test data 2.0 [LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University].

Eck, M., Vogel, S., & Waibel, A. (2004). Language model adaptation for statistical machine translation based on information retrieval. *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*.

Eck, M., Vogel, S., & Waibel, A. (2005). Low cost portability for statistical machine translation based on n-gram frequency and TF-IDF. *International Workshop on Spoken Language Translation (IWSLT) 2005*.

Edunov, S., Ott, M., Auli, M., Grangier, D. et al. (2018a). Classical structured prediction losses for sequence to sequence learning. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, *1*, 355–364.

Edunov, S., Ott, M., Auli, M., & Grangier, D. (2018b). Understanding back-translation at scale. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 489–500.

Eriguchi, A., Tsuruoka, Y., & Cho, K. (2017). Learning to parse and translate improves neural machine translation. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, *2*, 72–78.

Escudé Font, J., & Costa-jussà, M. R. (2019). Equalizing gender bias in neural machine translation with word embeddings techniques. *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, 147–154.

Farajian, M. A., Turchi, M., Negri, M., & Federico, M. (2017). Multi-domain neural machine translation through unsupervised adaptation. *Proceedings of the Second Conference on Machine Translation*, 127–137.

Federico, M. (2018). Challenges in adaptive neural machine translation. *Proceedings of the AMTA 2018 Workshop on Translation Quality Estimation and Automatic Post-Editing*, 207–242.

Feely, W., Hasler, E., & de Gispert, A. (2019). Controlling Japanese honorifics in English-to-Japanese neural machine translation. *Proceedings of the 6th Workshop on Asian Translation*, 45–53.

Firat, O., Sankaran, B., Al-onaizan, Y., Yarman Vural, F. T., & Cho, K. (2016). Zero-resource translation with multi-lingual neural machine translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 268–277.

Foster, G., Goutte, C., & Kuhn, R. (2010). Discriminative instance weighting for domain adaptation in statistical machine translation. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 451–459.

Frederking, R., & Nirenburg, S. (1994). Three heads are better than one. *Fourth Conference on Applied Natural Language Processing*, 95–100.

Freitag, M., & Al-Onaizan, Y. (2017). Beam search strategies for neural machine translation. *Proceedings of the First Workshop on Neural Machine Translation*, 56–60.

Freitag, M., & Al-Onaizan, Y. (2016). Fast domain adaptation for Neural Machine Translation. *CoRR*, *abs/1612.06897*.

Freitag, M., Al-Onaizan, Y., & Sankaran, B. (2017). Ensemble distillation for neural machine translation. *arXiv preprint arXiv:1702.01802*.

Friedman, B., & Nissenbaum, H. (1996). Bias in computer systems. *ACM Trans. Inf. Syst.*, *14*(3), 330–347.

Fukuda, T., Suzuki, M., Kurata, G., Thomas, S., Cui, J., & Ramabhadran, B. (2017). Efficient knowledge distillation from an ensemble of teachers. *INTERSPEECH*.

Gage, P. (1994). A new algorithm for data compression. *The C Users Journal*, *12*(2), 23–38.

Gallé, M. (2019). Investigating the effectiveness of BPE: The power of shorter sequences. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1375–1381.

Garmash, E., & Monz, C. (2016). Ensemble learning for multi-source Neural Machine Translation. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 1409–1418.

Gonen, H., & Goldberg, Y. (2019). Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 609–614.

González, A. V., Barrett, M., Hvingelby, R., Webster, K., & Søgaard, A. (2020). Type B reflexivization as an unambiguous testbed for multilingual multi-task gender bias. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2637–2648.

Goodfellow, I., Warde-Farley, D., Mirza, M., Courville, A., & Bengio, Y. (2013). Maxout networks. *International conference on machine learning*, 1319–1327.

Gordon, M., & Duh, K. (2020). Distill, adapt, distill: Training small, in-domain models for neural machine translation. *Proceedings of the Fourth Workshop on Neural Generation and Translation*, 110–118.

Green, S., & DeNero, J. (2012). A class-based agreement model for generating accurately inflected translations. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 146–155.

Greer, K., Lowerre, B., & Wilcox, L. (1982). Acoustic pattern matching and beam searching. *ICASSP'82. IEEE International Conference on Acoustics, Speech, and Signal Processing*, *7*, 1251–1254.

Gu, J., Bradbury, J., Xiong, C., Li, V. O., & Socher, R. (2018). Non-autoregressive neural machine translation. *International Conference on Learning Representations*.

Gu, S., Feng, Y., & Liu, Q. (2019). Improving domain adaptation translation with domain invariant and specific information. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 3081–3091.

Gulcehre, C., Firat, O., Xu, K., Cho, K., & Bengio, Y. (2017). On integrating a language model into neural machine translation. *Computer Speech and Language*, *45*(100), 137–148.

Gülçehre, C., Firat, O., Xu, K., Cho, K., Barrault, L., Lin, H.-C., Bougares, F., Schwenk, H., & Bengio, Y. (2015). On using monolingual corpora in Neural Machine Translation. *CoRR*, *abs/1503.03535*.

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, *12*(10), 993–1001.

Hasler, E. (2015). Dynamic topic adaptation for improved contextual modelling in statistical machine translation. *PhD Thesis*.

Hasler, E., de Gispert, A., Iglesias, G., & Byrne, B. (2018). Neural machine translation decoding with terminology constraints. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 506–512.

Hasler, E., de Gispert, A., Stahlberg, F., Waite, A., & Byrne, B. (2017). Source sentence simplification for statistical machine translation. *Computer Speech & Language*, *45*, 221–235.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. *Proceedings of the Sixth Workshop on Statistical Machine Translation*, 187–197.

Heafield, K., Pouzyrevsky, I., Clark, J. H., & Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 690–696.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Hinton, G. E., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *ArXiv*, *abs/1503.02531*.

HLEG, A. (2019). *Ethics guidelines for trustworthy AI*. High-Level Expert Group on Artificial Intelligence.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Hokamp, C. (2017). Ensembling factored neural machine translation models for automatic post-editing and quality estimation. *Proceedings of the Second Conference on Machine Translation*, 647–654.

Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with bloom embeddings. *Convolutional Neural Networks and Incremental Parsing*.

Hoosain, R. (1991). *Psycholinguistic implications for linguistic relativity: A case study of chinese*. Psychology Press.

Hord, L. C. (2016). Bucking the linguistic binary. *Western Papers in Linguistics*, *3*(1).

Hovy, D., Bianchi, F., & Fornaciari, T. (2020). "You Sound Just Like Your Father" Commercial Machine Translation Systems Include Stylistic Biases. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1686–1690.

Hu, J., Xia, M., Neubig, G., & Carbonell, J. (2019). Domain adaptation of neural machine translation by lexicon induction. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2989–3001.

Huck, M., Birch, A., & Haddow, B. (2015). Mixed-Domain vs. Multi-Domain Statistical Machine Translation. *Proceedings of the 15th Machine Translation Summit (MT-Summit 2015)*, 240–255.

Huck, M., Riess, S., & Fraser, A. (2017). Target-side word segmentation strategies for neural machine translation. *Proceedings of the Second Conference on Machine Translation*, 56–67.

Imamura, K., Fujita, A., & Sumita, E. (2018). Enhancement of encoder and attention using target monolingual corpora in neural machine translation. *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, 55–63.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*, 448–456.

Jean, S., Lauly, S., Firat, O., & Cho, K. (2017). Does neural machine translation benefit from larger context? *arXiv preprint arXiv:1704.05135*.

Jean, S., Cho, K., Memisevic, R., & Bengio, Y. (2015). On using very large target vocabulary for neural machine translation. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1–10.

Jean, S., Firat, O., Cho, K., Memisevic, R., & Bengio, Y. (2015). Montreal neural machine translation systems for WMT'15. *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 134–140.

Jimeno Yepes, A., Névéol, A., Neves, M., Verspoor, K., Bojar, O., Boyer, A., Grozea, C., Haddow, B., Kittner, M., Lichtblau, Y., Pecina, P., Roller, R., Rosa, R., Siu, A., Thomas, P., & Trescher, S. (2017). Findings of the WMT 2017 biomedical translation shared task. *Proceedings of the Second Conference on Machine Translation*, 234–247.

Johansen, A. R., Hansen, J. M., Obeid, E. K., Sønderby, C. K., & Winther, O. (2016). Neural machine translation with characters and hierarchical encoding. *arXiv preprint arXiv:1610.06550*.

Johnson, M. (2018). Providing gender-specific translations in Google Translate [(accessed: Aug 2020)].

Joty, S., Sajjad, H., Durrani, N., Al-Mannai, K., Abdelali, A., & Vogel, S. (2015). How to avoid unwanted pregnancies: Domain adaptation using neural network models. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1259–1270.

Junczys-Dowmunt, M. (2018a). Dual conditional cross-entropy filtering of noisy parallel corpora. *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 888–895.

Junczys-Dowmunt, M. (2019). Microsoft translator at WMT 2019: Towards large-scale document-level neural machine translation. *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 225–233.

Junczys-Dowmunt, M. (2018b). Microsoft's submission to the WMT2018 news translation task: How I learned to stop worrying and love the data. *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 425–430.

Junczys-Dowmunt, M., Dwojak, T., & Sennrich, R. (2016). The AMU-UEDIN submission to the WMT16 news translation task: Attention-based NMT models as feature functions in phrase-based SMT. *Proceedings of the First Conference on Machine Translation*, 319–325.

Kalchbrenner, N., & Blunsom, P. (2013). Recurrent continuous translation models. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1700–1709.

Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 655–665.

Karpinska, M., Li, B., Rogers, A., & Drozd, A. (2018). Subcharacter information in Japanese embeddings: When is it worth it? *Proceedings of the Workshop on the Relevance of Linguistic Structure in Neural Architectures for NLP*, 28–37.

Karpukhin, V., Levy, O., Eisenstein, J., & Ghazvininejad, M. (2019). Training on synthetic noise improves robustness to natural noise in machine translation. *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, 42–47.

Ke, Y., & Hagiwara, M. (2017). Radical-level ideograph encoder for rnn-based sentiment analysis of chinese and japanese. *Proceedings of Machine Learning Research*, *77*, 561–573.

Kessler, B., Nunberg, G., & Schutze, H. (1997). Automatic detection of text genre. *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, 32–38.

Khan, A., Panda, S., Xu, J., & Flokas, L. (2018). Hunter nmt system for wmt18 biomedical translation task: Transfer learning in neural machine translation. *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 655–661.

Khayrallah, H., & Koehn, P. (2018). On the impact of various types of noise on neural machine translation. *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, 74–83.

Khayrallah, H., Kumar, G., Duh, K., Post, M., & Koehn, P. (2017). Neural lattice search for domain adaptation in machine translation. *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 20–25.

Khayrallah, H., Thompson, B., Duh, K., & Koehn, P. (2018). Regularized Training Objective for Continued Training for Domain Adaptation in Neural Machine Translation. *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, 36–44.

Kim, Y., Jernite, Y., Sontag, D., & Rush, A. M. (2016). Character-aware neural language models. *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, 2741–2749.

Kim, Y., & Rush, A. M. (2016). Sequence-level knowledge distillation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1317–1327.

Kim, Y., Tran, D. T., & Ney, H. (2019). When and why is document-level context useful in neural machine translation? *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, 24–34.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations ICLR*.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A. et al. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, *114*(13), 3521–3526.

Kobus, C., Crego, J., & Senellart, J. (2017). Domain control for neural machine translation. *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, 372–378.

Kocmi, T., & Bojar, O. (2017). Curriculum learning and minibatch bucketing in neural machine translation. *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, 379–386.

Kocmi, T., & Bojar, O. (2018). Trivial transfer learning for low-resource neural machine translation. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 244–252.

Kocmi, T., Limisiewicz, T., & Stanovsky, G. (2020). Gender coreference and bias evaluation at wmt 2020. *Proceedings of the Fifth Conference on Machine Translation*.

Koehn, P., Duh, K., & Thompson, B. (2018). The JHU machine translation systems for WMT 2018. *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 438–444.

Koehn, P., & Knowles, R. (2017). Six challenges for neural machine translation. *Proceedings of the First Workshop on Neural Machine Translation*, 28–39.

Kornai, A. (2002). How many words are there? *Glottometrics*, *4*, 61–86.

Kothur, S. S. R., Knowles, R., & Koehn, P. (2018). Document-level adaptation for neural machine translation. *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, 64–73.

Kudo, T. (2018). Subword regularization: Improving neural network translation models with multiple subword candidates. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 66–75.

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71.

Kumar, G., Foster, G., Cherry, C., & Krikun, M. (2019). Reinforcement learning based curriculum optimization for neural machine translation. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2054–2061.

Kumar, S., & Byrne, W. (2004). Minimum Bayes-risk decoding for statistical machine translation. *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, 169–176.

Lambert, P., Schwenk, H., Servan, C., & Abdul-Rauf, S. (2011). Investigations on translation model adaptation using monolingual data. *Proceedings of the Sixth Workshop on Statistical Machine Translation*, 284–293.

Le, H. S., Allauzen, A., & Yvon, F. (2012). Continuous space translation models with neural networks. *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 39–48.

LeCun, Y., Denker, J. S., & Solla, S. A. (1990). Optimal brain damage. *Advances in neural information processing systems*, 598–605.

Lee, J., Cho, K., & Hofmann, T. (2017). Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, *5*, 365–378.

Lee, Y.-B., & Myaeng, S. H. (2002). Text genre classification with genre-revealing and subject-revealing features. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 145–150.

Levin, E., & Fleisher, M. (1988). Accelerated learning in layered neural networks. *Complex Systems*, *2*, 625–640.

Lewis, W., & Eetemadi, S. (2013). Dramatically reducing training data size through vocabulary saturation. *Proceedings of the Eighth Workshop on Statistical Machine Translation*, 281–291.

Li, J., & Jurafsky, D. (2016). Mutual information and diverse decoding improve neural machine translation. *arXiv preprint arXiv:1601.00372*.

Li, X., Zhang, J., & Zong, C. (2018). One sentence one model for neural machine translation. *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Li, X., Zhang, J., & Zong, C. (2016). Towards zero unknown word in neural machine translation. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 2852–2858.

Li, Z., Wang, R., Chen, K., Utiyama, M., Sumita, E., Zhang, Z., & Zhao, H. (2020). Explicit sentence compression for neural machine translation. *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI*, 8311–8318.

Lin, C.-Y., & Och, F. J. (2004). ORANGE: A method for evaluating automatic evaluation metrics for machine translation. *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, 501–507.

Ling, W., Trancoso, I., Dyer, C., & Black, A. W. (2015). Character-based neural machine translation. *arXiv preprint arXiv:1511.04586*.

Liu, L., Utiyama, M., Finch, A., & Sumita, E. (2016). Agreement on target-bidirectional neural machine translation. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 411–416.

Liu, P. D., Chung, K. K., McBride-Chang, C., & Tong, X. (2010). Holistic versus analytic processing: Evidence for a different approach to processing of Chinese at the word and character levels in Chinese children. *Journal of Experimental Child Psychology*, *107*(4), 466–478.

Liu, X., Lai, H., Wong, D. F., & Chao, L. S. (2020). Norm-based curriculum learning for neural machine translation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 427–436.

Liu, X., Chen, X., Wang, Y., Gales, M. J., & Woodland, P. C. (2016). Two efficient lattice rescoring methods using recurrent neural network language models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *24*(8), 1438–1449.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Liu, Y., Zhou, L., Wang, Y., Zhao, Y., Zhang, J., & Zong, C. (2018). A comparable study on model averaging, ensembling and reranking in nmt. *CCF International Conference on Natural Language Processing and Chinese Computing*, 299–308.

Lu, K., Mardziel, P., Wu, F., Amancharla, P., & Datta, A. (2020). Gender bias in neural natural language processing. *12300*, 189–202.

Luong, M.-T., & Manning, C. D. (2016). Achieving open vocabulary neural machine translation with hybrid word-character models. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1054–1063.

Luong, M.-T., & Manning, C. D. (2015). Stanford Neural Machine Translation systems for spoken language domains. *Proceedings of the International Workshop on Spoken Language Translation*, 76–79.

Luong, T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1412–1421.

Luong, T., Sutskever, I., Le, Q., Vinyals, O., & Zaremba, W. (2015). Addressing the rare word problem in neural machine translation. *Proceedings of the 53rd Annual Meeting*

of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 11–19.

Macé, V., & Servan, C. (2019). Using whole document context in neural machine translation. *arXiv preprint arXiv:1910.07481*.

Macháček, D., Vidra, J., & Bojar, O. (2018). Morphological and language-agnostic word segmentation for nmt. *International Conference on Text, Speech, and Dialogue*, 277–284.

Manzini, T., Yao Chong, L., Black, A. W., & Tsvetkov, Y. (2019). Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 615–621.

McCandlish, S., Kaplan, J., Amodei, D., & Team, O. D. (2018). An empirical model of large-batch training. *arXiv preprint arXiv:1812.06162*.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of learning and motivation* (pp. 109–165). Elsevier.

Mehta, S., Azarnoush, B., Chen, B., Saluja, A., Misra, V., Bihani, B., & Kumar, R. (2020). Simplify-then-translate: Automatic prep rocessing for black-box machine translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*.

Mei, Y. (1615). *Zi hui*.

Mghabbar, I., & Ratnamogan, P. (2020). Building a multi-domain neural machine translation model using knowledge distillation. *ECAI 2020 - 24th European Conference on Artificial Intelligence*, *325*, 2116–2123.

Michel, P., & Neubig, G. (2018). Extreme adaptation for personalized neural machine translation. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 312–318.

Miculicich Werlen, L., & Popescu-Belis, A. (2017). Using coreference links to improve Spanish-to-English machine translation. *Proceedings of the 2nd Workshop on Coreference Resolution Beyond OntoNotes (CORBON 2017)*, 30–40.

Mikolov, T. (2012). Statistical language models based on neural networks. *PhD Thesis*.

Mikolov, T., Corrado, G., Chen, K., & Dean, J. (2013). Efficient estimation of word representations in vector space. *ICLR*.

Mikolov, T., Karafiát, M., Burget, L., Černocký, J., & Khudanpur, S. (2010). Recurrent neural network based language model. *Eleventh annual conference of the international speech communication association*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 3111–3119.

Misersky, J., Majid, A., & Snijders, T. M. (2019). Grammatical gender in German influences how role-nouns are interpreted: Evidence from ERPs. *Discourse Processes*, *56*(8), 643–654.

Mizumoto, T., Hayashibe, Y., Komachi, M., Nagata, M., & Matsumoto, Y. (2012). The effect of learner corpus size in grammatical error correction of ESL writings. *Proceedings of COLING 2012: Posters*, 863–872.

Moore, R. C., & Lewis, W. (2010). Intelligent selection of language model training data. *Proceedings of the ACL 2010 Conference Short Papers*, 220–224.

Morishita, M., Oda, Y., Neubig, G., Yoshino, K., Sudoh, K., & Nakamura, S. (2017). An empirical study of mini-batch creation strategies for neural machine translation. *Proceedings of the First Workshop on Neural Machine Translation*, 61–68.

Morishita, M., Suzuki, J., & Nagata, M. (2017). NTT Neural Machine Translation Systems at WAT 2017. *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, 89–94.

Moryossef, A., Aharoni, R., & Goldberg, Y. (2019). Filling gender & number gaps in neural machine translation with black-box context injection. *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, 49–54.

Müller, M., Rios, A., & Sennrich, R. (2020). Domain robustness in neural machine translation. *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (AMTA 2020)*, 151–164.

Murray, K., & Chiang, D. (2018). Correcting length bias in neural machine translation. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 212–223.

Nadejde, M., Reddy, S., Sennrich, R., Dwojak, T., Junczys-Dowmunt, M., Koehn, P., & Birch, A. (2017). Predicting target language CCG supertags improves neural machine translation. *Proceedings of the Second Conference on Machine Translation*, 68–79.

Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., & Isahara, H. (2016). ASPEC: Asian Scientific Paper Excerpt Corpus. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, 2204–2208.

Nakov, P., Guzman, F., & Vogel, S. (2012). Optimizing for sentence-level BLEU+1 yields short translations. *Proceedings of COLING 2012*, 1979–1994.

Napoles, C., Sakaguchi, K., Post, M., & Tetreault, J. (2015). Ground truth for grammatical error correction metrics. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 588–593.

Napoles, C., Sakaguchi, K., & Tetreault, J. (2017). JFLEG: A fluency corpus and benchmark for grammatical error correction. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 229–234.

Needleman, M. (2000). The unicode standard. *Serials review*, *26*(2), 51–54.

Neishi, M., Sakuma, J., Tohda, S., Ishiwatari, S., Yoshinaga, N., & Toyoda, M. (2017). A bag of useful tricks for practical neural machine translation: Embedding layer initialization and large batch size. *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, 99–109.

Neubig, G. (2011). The Kyoto free translation task.

Neubig, G. (2016). Lexicons and minimum risk training for neural machine translation: NAIST-CMU at WAT2016. *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, 119–125.

Neves, M. L., Jimeno-Yepes, A., & Névéol, A. (2016). The ScieLO Corpus: a Parallel Corpus of Scientific Publications for Biomedicine. *LREC*.

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H., & Bryant, C. (2014). The CoNLL-2014 shared task on grammatical error correction. *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, 1–14.

Nguyen, K., Daumé III, H., & Boyd-Graber, J. (2017). Reinforcement learning for bandit neural machine translation with simulated human feedback. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1464–1474.

Nguyen, T., & Chiang, D. (2018). Improving lexical choice in neural machine translation. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 334–343.

Nguyen, V., Brooke, J., & Baldwin, T. (2017). Sub-character neural language modelling in Japanese. *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, 148–153.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, 160–167.

Och, F. J., & Weber, H. (1998). Improving statistical natural language translation with categories and rules. *COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics*.

Oda, Y., Neubig, G., Sakti, S., Toda, T., & Nakamura, S. (2015). Ckylark: A more robust PCFG-LA parser. *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, 41–45.

Papadopoulos, B. (2019). *Innovaciones al género morfológico en el español de hablantes genderqueer (Morphological gender innovations in Spanish of genderqueer speakers)*. eScholarship, University of California.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311–318.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. *International conference on machine learning*, 1310–1318.

Patrias, K., & Wendling, D. (2007). Citing medicine: The nlm style guide for authors, editors. *and publishers. Bethesda, MD: National Library of Medicine. Retrieved June*, *27*, 2011.

Pereyra, G., Tucker, G., Chorowski, J., Kaiser, Ł., & Hinton, G. E. (2017). Regularizing neural networks by penalizing confident output distributions. *5th International Conference on Learning Representations, ICLR*.

Pham, M. Q., Crego, J.-M., Yvon, F., & Senellart, J. (2019). Generic and specialized word embeddings for multi-domain machine translation. *International Workshop on Spoken Language Translation*.

Platanios, E. A., Stretcu, O., Neubig, G., Poczos, B., & Mitchell, T. (2019). Competence-based curriculum learning for neural machine translation. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1162–1172.

Poncelas, A., Maillette de Buy Wenniger, G., & Way, A. (2019). Transductive data-selection algorithms for fine-tuning neural machine translation. *Proceedings of The 8th Workshop on Patent and Scientific Literature Translation*, 13–23.

Poncelas, A., Shterionov, D., Way, A., de Buy Wenniger, G. M., & Passban, P. (2018). Investigating backtranslation in neural machine translation. *Proceedings of the 21st Annual Conference of the European Association for Machine Translation: 28-30 May 2018, Universitat d'Alacant, Alacant, Spain*, 249–258.

Poncelas, A., & Way, A. (2019). Selecting artificially-generated sentences for fine-tuning neural machine translation. *Proceedings of the 12th International Conference on Natural Language Generation*, 219–228.

Poncelas, A., Wenniger, G. M. d. B., & Way, A. (2018). Data selection with feature decay algorithms using an approximated target side. *IWSLT 2018, International Workshop on Spoken Language Translation*.

Popel, M., & Bojar, O. (2018). Training tips for the transformer model. *The Prague Bulletin of Mathematical Linguistics*, *110*(1), 43–70.

Post, M. (2018). A call for clarity in reporting BLEU scores. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 186–191.

Pouget-Abadie, J., Bahdanau, D., van Merriënboer, B., Cho, K., & Bengio, Y. (2014). Overcoming the curse of sentence length for neural machine translation using automatic segmentation. *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, 78–85.

Prates, M. O., Avelar, P. H., & Lamb, L. C. (2019). Assessing gender bias in machine translation: A case study with google translate. *Neural Computing and Applications*, 1–19.

Puurtinen, T. (2003). Explicitating and implicitating source text ideology. *Across Languages and Cultures*, *4*(1), 53–62.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners.

Ragni, A., Saunders, D., Zahemszky, P., Vasilakes, J., Gales, M. J. F., & Knill, K. M. (2017). Morph-to-word transduction for accurate and efficient automatic speech recognition and keyword search. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5770–5774.

Ranzato, M., Chopra, S., Auli, M., & Zaremba, W. (2016). Sequence level training with recurrent neural networks. *ICLR*.

Ratcliff, R. (1990). Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions. *Psychological review*, *97*(2), 285.

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review*, *33*(1-2), 1–39.

Rosti, A.-V., Ayan, N. F., Xiang, B., Matsoukas, S., Schwartz, R., & Dorr, B. (2007). Combining outputs from multiple machine translation systems. *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, 228–235.

Rudinger, R., Naradowsky, J., Leonard, B., & Van Durme, B. (2018). Gender bias in coreference resolution. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 8–14.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, *323*(6088), 533–536.

Sajjad, H., Durrani, N., Dalvi, F., Belinkov, Y., & Vogel, S. (2017). Neural Machine Translation training in a multi-domain scenario. *IWSLT 2017, International Workshop on Spoken Language Translation*.

Sakaguchi, K., Post, M., & Van Durme, B. (2017). Grammatical error correction with neural reinforcement learning. *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 366–372.

Salesky, E., Runge, A., Coda, A., Niehues, J., & Neubig, G. (2020). Optimizing segmentation granularity for neural machine translation. *Machine Translation*, 1–19.

Sallis, R. (2021). Using training data to reduce gender bias in neural machine translation. *MEng Thesis, Cambridge University Engineering Department*.

Santamaría, L., & Axelrod, A. (2017). Data selection with cluster-based language difference models and cynical selection. *IWSLT 2017, International Workshop on Spoken Language Translation*.

Santini, M. (2004). State-of-the-art on automatic genre identification. *Technical report*.

Santurkar, S., Tsipras, D., Ilyas, A., & Madry, A. (2018). How does batch normalization help optimization? *Advances in Neural Information Processing Systems*, 2483–2493.

Saunders, D., & Byrne, B. (2020a). Addressing Exposure Bias With Document Minimum Risk Training: Cambridge at the WMT20 Biomedical Translation Task. *Proceedings of the Fifth Conference on Machine Translation*, 862–869.

Saunders, D., & Byrne, B. (2020b). Reducing gender bias in neural machine translation as a domain adaptation problem. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7724–7736.

Saunders, D., Feely, W., & Byrne, B. (2020). Inference-only sub-character decomposition improves translation of unseen logographic characters. *Proceedings of the 7th Workshop on Asian Translation*, 170–177.

Saunders, D., Sallis, R., & Byrne, B. (2020). Neural machine translation doesn't translate gender coreference right unless you make it. *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, 35–43.

Saunders, D., Stahlberg, F., & Byrne, B. (2019). UCAM biomedical translation at WMT19: Transfer learning multi-domain ensembles. *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, 169–174.

Saunders, D., Stahlberg, F., & Byrne, B. (2020). Using context in neural machine translation training objectives. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7764–7770.

Saunders, D., Stahlberg, F., de Gispert, A., & Byrne, B. (2019). Domain adaptive inference for neural machine translation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 222–228.

Saunders, D., Stahlberg, F., de Gispert, A., & Byrne, B. (2018). Multi-representation ensembles and delayed SGD updates improve syntax-based NMT. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 319–325.

Savenkov, K. (2018). State of the domain-adaptive machine translation [(accessed: Nov 2020)].

Schamper, J., Rosendahl, J., Bahar, P., Kim, Y., Nix, A., & Ney, H. (2018). The RWTH Aachen University supervised machine translation systems for WMT 2018. *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 496–503.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, *45*(11), 2673–2681.

Schwarzwald, O. (1982). Feminine formation in modern Hebrew. *Hebrew Annual Review*, *6*, 153–178.

Schwenk, H. (2012). Continuous space translation models for phrase-based statistical machine translation. *Proceedings of COLING 2012: Posters*, 1071–1080.

Sennrich, R., Birch, A., Currey, A., Germann, U., Haddow, B., Heafield, K., Barone, A. V. M., & Williams, P. (2017). The university of Edinburgh's neural MT systems for WMT17. *Proceedings of the Second Conference on Machine Translation*, 389–399.

Sennrich, R., & Haddow, B. (2016). Linguistic input features improve neural machine translation. *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, 83–91.

Sennrich, R., Haddow, B., & Birch, A. (2016a). Controlling politeness in neural machine translation via side constraints. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 35–40.

Sennrich, R., Haddow, B., & Birch, A. (2016b). Edinburgh neural machine translation systems for WMT 16. *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, 371–376.

Sennrich, R., Haddow, B., & Birch, A. (2016c). Improving Neural Machine Translation Models with Monolingual Data. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, *1*, 86–96.

Sennrich, R., Haddow, B., & Birch, A. (2016d). Neural machine translation of rare words with subword units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, *1*, 1715–1725.

Sennrich, R., & Zhang, B. (2019). Revisiting low-resource neural machine translation: A case study. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 211–221.

Shah, D. S., Schwartz, H. A., & Hovy, D. (2020). Predictive biases in natural language processing models: A conceptual framework and overview. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5248–5264.

Shannon, M. (2017). Optimizing expected word error rate via sampling for speech recognition. *Proc. Interspeech 2017*, 3537–3541.

Shen, S., Cheng, Y., He, Z., He, W., Wu, H., Sun, M., & Liu, Y. (2016). Minimum Risk Training for Neural Machine Translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, *1*, 1683–1692.

Shroy, A. J. (2016). Innovations in gender-neutral French: Language practices of nonbinary French speakers on Twitter. *Ms., University of California, Davis*.

Shtrikman, S. (1994). Some comments on zipf's law for the chinese language. *Journal of Information Science*, *20*(2), 142–143.

Sim, K. C., Byrne, W. J., Gales, M. J., Sahbi, H., & Woodland, P. C. (2007). Consensus network decoding for statistical machine translation system combination. *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, *4*, IV–105.

Sinclair, J., & Ball, J. (1996). Preliminary recommendations on text typology. *EAGLES (Expert Advisory Group on Language Engineering Standards)*.

Smith, S. L., Kindermans, P.-J., & Le, Q. V. (2018). Don't decay the learning rate, increase the batch size. *International Conference on Learning Representations*.

Snover, M. (2006). A study of translation edit rate with targeted human annotation. *Proc. Association for Machine Translation in the Americas (AMTA2006)*.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., & Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 151–161.

Song, K., Zhang, Y., Yu, H., Luo, W., Wang, K., & Zhang, M. (2019). Code-switching for enhancing NMT with pre-specified translation. *Proceedings of the 2019 Conference*

of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 449–459.

Song, X., Cohn, T., & Specia, L. (2013). Bleu deconstructed: Designing a better mt evaluation metric. *International Journal of Computational Linguistics and Applications*, *4*(2), 29–44.

Sriram, A., Jun, H., Satheesh, S., & Coates, A. (2018). Cold fusion: Training seq2seq models together with language models. *Proc. Interspeech 2018*, 387–391.

Stafanovičs, A., Bergmanis, T., & Pinnis, M. (2020). Mitigating gender bias in machine translation with target gender annotations. *Proceedings of the Fifth Conference on Machine Translation (WMT)*.

Stahlberg, F., Bryant, C., & Byrne, B. (2019). Neural grammatical error correction with finite state transducers. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4033–4039.

Stahlberg, F., & Byrne, B. (2019). On NMT search errors and model errors: Cat got your tongue? *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3356–3362.

Stahlberg, F., & Byrne, B. (2017). Unfolding and shrinking neural machine translation ensembles. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1946–1956.

Stahlberg, F., Cross, J., & Stoyanov, V. (2018). Simple fusion: Return of the language model. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 204–211.

Stahlberg, F., de Gispert, A., & Byrne, B. (2018). The university of Cambridge's machine translation systems for WMT18. *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 504–512.

Stahlberg, F., de Gispert, A., Hasler, E., & Byrne, B. (2017). Neural machine translation by minimising the Bayes-risk with respect to syntactic translation lattices. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 362–368.

Stahlberg, F., Hasler, E., Saunders, D., & Byrne, B. (2017). SGNMT – a flexible NMT decoding platform for quick prototyping of new models and search strategies. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 25–30.

Stahlberg, F., Hasler, E., Waite, A., & Byrne, B. (2016). Syntactically guided neural machine translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 299–305.

Stahlberg, F., Saunders, D., de Gispert, A., & Byrne, B. (2019). CUED@WMT19:EWC&LMs. *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 364–373.

Stahlberg, F., Saunders, D., Iglesias, G., & Byrne, B. (2018). Why not be versatile? applications of the SGNMT decoder for machine translation. *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, 208–216.

Stanovsky, G., Smith, N. A., & Zettlemoyer, L. (2019). Evaluating gender bias in machine translation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1679–1684.

Sun, T., Gaut, A., Tang, S., Huang, Y., ElSherief, M., Zhao, J., Mirza, D., Belding, E., Chang, K.-W., & Wang, W. Y. (2019). Mitigating gender bias in natural language processing: Literature review. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1630–1640.

Sun, Y., Lin, L., Yang, N., Ji, Z., & Wang, X. (2014). Radical-enhanced chinese character embedding. *International Conference on Neural Information Processing*, 279–286.

Sutskever, I., Martens, J., & Hinton, G. E. (2011). Generating text with recurrent neural networks. *Proceedings of the 28th International Conference on International Conference on Machine Learning*, 1017–1024.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 3104–3112.

Sutton, R. S., McAllester, D. A., Singh, S. P., & Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 1057–1063.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.

Taghipour, K., Khadivi, S., & Xu, J. (2011). Parallel corpus refinement as an outlier detection algorithm. *Proceedings of the 13th Machine Translation Summit (MT Summit XIII)*, 414–421.

Tamchyna, A., Weller-Di Marco, M., & Fraser, A. (2017). Modeling target-side inflection in neural machine translation. *Proceedings of the Second Conference on Machine Translation*, 32–42.

Tan, S., Joty, S., Kan, M.-Y., & Socher, R. (2020). It's morphin' time! Combating linguistic discrimination with inflectional perturbations. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2920–2935.

Tars, S., & Fishel, M. (2018). Multi-domain neural machine translation. *Proceedings of the 21st Annual Conference of the European Association for Machine Translation: 28-30 May 2018, Universitat d'Alacant, Alacant, Spain*, 259–268.

Tebbifakhr, A., Agrawal, R., Negri, M., & Turchi, M. (2018). Multi-source transformer with combined losses for automatic post editing. *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 846–852.

Thompson, B., Gwinnup, J., Khayrallah, H., Duh, K., & Koehn, P. (2019). Overcoming catastrophic forgetting during domain adaptation of neural machine translation. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Thompson, B., Khayrallah, H., Anastasopoulos, A., McCarthy, A. D., Duh, K., Marvin, R., McNamee, P., Gwinnup, J., Anderson, T., & Koehn, P. (2018). Freezing subnetworks to analyze domain adaptation in Neural Machine Translation. *Proceedings of the Third Conference on Machine Translation*, 124–132.

Tiedemann, J., Scherrer, Y. et al. (2017). Neural machine translation with extended context. *Proceedings of the Third Workshop on Discourse in Machine Translation*.

Tomalin, M., Byrne, B., Concannon, S., Saunders, D., & Ullmann, S. (2021). The Practical Ethics of Bias Reduction in Machine Translation: Why Domain Adaptation is Better than Data Debiasing. *Ethics and Information Technology*.

Tromble, R., Kumar, S., Och, F., & Macherey, W. (2008). Lattice Minimum Bayes-Risk decoding for statistical machine translation. *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 620–629.

Tsarfaty, R., Sadde, S., Klein, S., & Seker, A. (2019). What's wrong with Hebrew NLP? and how to make it right. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, 259–264.

Tu, Z., Lu, Z., Liu, Y., Liu, X., & Li, H. (2016). Modeling coverage for neural machine translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 76–85.

Turchi, M., Chatterjee, R., & Negri, M. (2017). WMT17 en-de APE shared task data [LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University].

Turian, J., Ratinov, L.-A., & Bengio, Y. (2010). Word representations: A simple and general method for semi-supervised learning. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 384–394.

Vaibhav, V., Singh, S., Stewart, C., & Neubig, G. (2019). Improving robustness of machine translation with synthetic noise. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1916–1920.

van der Wees, M. (2017). What's in a domain?: Towards fine-grained adaptation for machine translation. *PhD Thesis*.

van der Wees, M., Bisazza, A., & Monz, C. (2017). Dynamic data selection for neural machine translation. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1400–1410.

Vanmassenhove, E., Hardmeier, C., & Way, A. (2018). Getting gender right in neural machine translation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3003–3008.

Vaswani, A., Bengio, S., Brevdo, E., Chollet, F., Gomez, A. N., Gouws, S., Jones, L., Kaiser, Ł., Kalchbrenner, N., Parmar, N., Sepassi, R., Shazeer, N., & Uszkoreit, J. (2018). Tensor2Tensor for Neural Machine Translation. *CoRR*, *abs/1803.07416*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 6000–6010.

Vaswani, A., Zhao, Y., Fossum, V., & Chiang, D. (2013). Decoding with large-scale neural language models improves translation. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1387–1392.

Vijayakumar, A. K., Cogswell, M., Selvaraju, R. R., Sun, Q., Lee, S., Crandall, D., & Batra, D. (2016). Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*.

Vilar, D. (2018). Learning hidden unit contribution for adapting neural machine translation models. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, *2*, 500–505.

Voita, E., Sennrich, R., & Titov, I. (2019). When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1198–1212.

Voita, E., Serdyukov, P., Sennrich, R., & Titov, I. (2018). Context-aware neural machine translation learns anaphora resolution. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1264–1274.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., & Titov, I. (2019). Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5797–5808.

Wang, C., Cho, K., & Gu, J. (2020). Neural machine translation with byte-level subwords, 9154–9160.

Wang, C., & Sennrich, R. (2020). On exposure bias, hallucination and domain shift in neural machine translation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 3544–3552.

Wang, L., Tu, Z., Way, A., & Liu, Q. (2017). Exploiting cross-sentence context for neural machine translation. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2826–2831.

Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D. F., & Chao, L. S. (2019). Learning deep transformer models for machine translation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1810–1822.

Wang, R., Finch, A., Utiyama, M., & Sumita, E. (2017). Sentence embedding for Neural Machine Translation domain adaptation. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, *2*, 560–566.

Wang, R., Utiyama, M., Liu, L., Chen, K., & Sumita, E. (2017). Instance weighting for neural machine translation domain adaptation. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1482–1488.

Wang, R., Utiyama, M., & Sumita, E. (2018). Dynamic sentence sampling for efficient training of neural machine translation. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 298–304.

Wang, W., Tian, Y., Ngiam, J., Yang, Y., Caswell, I., & Parekh, Z. (2020). Learning a multi-domain curriculum for neural machine translation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7711–7723.

Wang, W., Watanabe, T., Hughes, M., Nakagawa, T., & Chelba, C. (2018). Denoising neural machine translation training with trusted data and online data selection. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 133–143.

Wang, Y., Wang, L., Shi, S., Li, V. O. K., & Tu, Z. (2020). Go from the general to the particular: Multi-domain translation with domain transformation networks. *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI*, 9233–9241.

Watanabe, T., Suzuki, J., Tsukada, H., & Isozaki, H. (2007). Online large-margin training for statistical machine translation. *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 764–773.

Wei, H.-R., Zhang, Z., Chen, B., & Luo, W. (2020). Iterative domain-repaired back-translation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5884–5893.

Weinshall, D., Cohen, G., & Amir, D. (2018). Curriculum learning by transfer learning: Theory and experiments with deep networks. *International Conference on Machine Learning*, 5238–5246.

Welleck, S., Brantley, K., Daumé, H., & Cho, K. (2019). Non-monotonic sequential text generation. *36th International Conference on Machine Learning, ICML 2019*, 11656–11676.

Werbos, P. J. (1988). Generalization of backpropagation with application to a recurrent gas market model. *Neural networks*, (4), 339–356.

Wieting, J., Berg-Kirkpatrick, T., Gimpel, K., & Neubig, G. (2019). Beyond BLEU:training neural machine translation with semantic similarity. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4344–4355.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, *8*(3-4), 229–256.

Wu, L., Tian, F., Qin, T., Lai, J., & Liu, T.-Y. (2018). A study of reinforcement learning for neural machine translation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3612–3621.

Wu, L., Zhao, L., Qin, T., Lai, J., & Liu, T.-Y. (2017). Sequence prediction with unlabeled data by reward function learning. *IJCAI*, 3098–3104.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K. et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Wuebker, J., Simianer, P., & DeNero, J. (2018). Compact personalized models for neural machine translation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 881–886.

Xie, Z., Avati, A., Arivazhagan, N., Jurafsky, D., & Ng, A. Y. (2016). Neural language correction with character-based attention. *arXiv preprint arXiv:1603.09727*.

Xu, J., Crego, J., & Senellart, J. (2019). Lexical micro-adaptation for neural machine translation. *International Workshop on Spoken Language Translation*.

Yushu, Z., Tingjing, C. et al. (1716). Kangxi zidian.

Zaremba, W., Sutskever, I., & Vinyals, O. (2014). Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Zeiler, M. D. (2012). Adadelta: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zeng, J., Su, J., Wen, H., Liu, Y., Xie, J., Yin, Y., & Zhao, J. (2018). Multi-domain neural machine translation with word-level domain context discrimination. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 447–457.

Zhang, D., Kim, J., Crego, J., & Senellart, J. (2017). Boosting neural machine translation. *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 271–276.

Zhang, J., & Zong, C. (2016). Bridging neural machine translation and bilingual dictionaries. *arXiv preprint arXiv:1610.07272*.

Zhang, J., Utiyama, M., Sumita, E., Neubig, G., & Nakamura, S. (2018). Guiding neural machine translation with retrieved translation pieces. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1325–1335.

Zhang, L., & Komachi, M. (2019). Chinese–Japanese unsupervised neural machine translation using sub-character level information. *Proceedings of the 33rd Pacific Asia Conference on Language, Information and Computation*, 309–315.

Zhang, L., & Komachi, M. (2018). Neural machine translation of logographic language using sub-character level information. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 17–25.

Zhang, S., & Xiong, D. (2018). Sentence weighting for neural machine translation domain adaptation. *Proceedings of the 27th International Conference on Computational Linguistics*, 3181–3190.

Zhang, X., Kumar, G., Khayrallah, H., Murray, K., Gwinnup, J., Martindale, M. J., McNamee, P., Duh, K., & Carpuat, M. (2018). An empirical exploration of curriculum learning for neural machine translation. *arXiv preprint arXiv:1811.00739*.

Zhang, X., Shapiro, P., Kumar, G., McNamee, P., Carpuat, M., & Duh, K. (2019). Curriculum learning for domain adaptation in neural machine translation. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1903–1915.

Zhang, Y., & Clark, S. (2011). Syntax-based grammaticality improvement using CCG and guided search. *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 1147–1157.

Zhang, Z., Wu, S., Liu, S., Li, M., Zhou, M., & Xu, T. (2019). Regularizing neural machine translation by target-bidirectional agreement. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*, 443–450.

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K.-W. (2018). Gender bias in coreference resolution: Evaluation and debiasing methods. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 15–20.

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K.-W. (2017). Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2979–2989.

Zhao, M., Wu, H., Niu, D., & Wang, X. (2020). Reinforced curriculum learning on pre-trained neural machine translation models. *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI*, 9652–9659.

Zhao, Y., Zhang, J., He, Z., Zong, C., & Wu, H. (2018). Addressing troublesome words in neural machine translation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 391–400.

Ziemski, M., Junczys-Dowmunt, M., & Pouliquen, B. (2016). The uni ted nations parallel corpus v1.0. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, 3530–3534.

Zimman, L. (2017). Transgender language reform: Some challenges and strategies for promoting trans-affirming, gender-inclusive language. *Journal of Language and Discrimination*, *1*(1), 83–104.

Zipf, G. K. (1949). Human behavior and the principle of least effort.

Zmigrod, R., Mielke, S. J., Wallach, H., & Cotterell, R. (2019). Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1651–1661.

Zoph, B., Yuret, D., May, J., & Knight, K. (2016). Transfer learning for low-resource neural machine translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1568–1575.

Appendix A

List of publications

Some work in this thesis draws from the following publications I authored or co-authored during my PhD:

Tomalin, M., Byrne, B., Concannon, S., **Saunders, D**, & Ullmann, S. (2021). The Practical Ethics of Bias Reduction in Machine Translation: Why Domain Adaptation is Better than Data Debiasing. *Ethics and Information Technology*.

Saunders, D, & Byrne, B. (2020a). Addressing Exposure Bias With Document Minimum Risk Training: Cambridge at the WMT20 Biomedical Translation Task. *Proceedings of the Fifth Conference on Machine Translation*, 862–869.

Saunders, D, & Byrne, B. (2020b). Reducing gender bias in neural machine translation as a domain adaptation problem. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7724–7736.

Saunders, D, Feely, W., & Byrne, B. (2020). Inference-only sub-character decomposition improves translation of unseen logographic characters. *Proceedings of the 7th Workshop on Asian Translation*, 170–177.

Saunders, D, Sallis, R., & Byrne, B. (2020). Neural machine translation doesn't translate gender coreference right unless you make it. *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, 35–43.

Saunders, D, Stahlberg, F., & Byrne, B. (2020). Using Context in Neural Machine Translation Training Objectives. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7764–7770.

Saunders, D, Stahlberg, F., & Byrne, B. (2019). UCAM Biomedical Translation at WMT19: Transfer Learning Multi-domain Ensembles. *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, 169–174.

Saunders, D, Stahlberg, F., de Gispert, A., & Byrne, B. (2019). Domain Adaptive Inference for Neural Machine Translation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 222–228.

Stahlberg, F., **Saunders, D**, de Gispert, A., & Byrne, B. (2019). CUED@WMT19:EWC&LMs. *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 364–373.

Saunders, D, Stahlberg, F., de Gispert, A., & Byrne, B. (2018). Multi-representation ensembles and delayed SGD updates improve syntax-based NMT. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 319–325.

Stahlberg, F., **Saunders, D**, Iglesias, G., & Byrne, B. (2018). Why not be versatile? applications of the SGNMT decoder for machine translation. *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, 208–216.

Stahlberg, F., Hasler, E., **Saunders, D**, & Byrne, B. (2017). SGNMT – a flexible NMT decoding platform for quick prototyping of new models and search strategies. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 25–30.